PHYSICS-DRIVEN VARIATIONAL METHODS FOR COMPUTER VISION AND SHAPE-BASED IMAGING

A Dissertation
Presented to
The Academic Faculty

by

Martin F. Mueller

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
August 2014

PHYSICS-DRIVEN VARIATIONAL METHODS FOR COMPUTER VISION AND SHAPE-BASED IMAGING

Approved by:

Dr. Anthony J. Yezzi, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Francesco G. Fedele, Co-Advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Patricio A. Vela
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Laurence J. Jacobs
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Justin K. Romberg
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Sung Ha Kang
School of Mathematics
Georgia Institute of Technology

Date Approved: May 1, 2014

To Mandy

for sharing the bumpy ride

ACKNOWLEDGEMENTS

First of all, thank you, Dr. Yezzi, for being my Ph.D. advisor throughout the final years of my Ph.D. I am deeply grateful about you and Dr. Fedele adopting the "last member" of the Tannenbaum group. These chapters would look very differently or would not have been written without your support in these difficult times. I want to thank Dr. Yezzi, Dr. Fedele, and Dr. Tannenbaum for guiding my research at different stages of my Ph.D. Dr. Tannenbaum introduced me to computer vision in my first year's flame detection project. Dr. Fedele is the creative mind behind the adjoint active contours for shape-based inverse scattering, and I am grateful for the interdisciplinary insight I got from this project. Finally, Dr. Yezzi gave me the opportunity to do research on the coupled photo-geometric object features in the final year of my Ph.D. I was lucky enough to collaborate with three outstanding minds in their fields.

Thank you, Dr. Jacobs, for your continued service as a mentor throughout my time at Georgia Tech. Your generous support and sincere consideration for your students were two of the reasons for me to come back to Tech for my Ph.D. studies. You were always available to talk, in good and in difficult times; that will not be forgotten.

Another round of thanks goes to my thesis committee members: Dr. Vela, Dr. Romberg, and Dr. Kang. Thank you for your service and your helpful comments on this dissertation.

Many fellow students accompanied me on this journey. From the—now officially

historic—MINERVA lab, I would like to mention in particular Ivan, Peter, and Liang-Jia. Together we concluded the chapter of MINERVA at Georgia Tech after Dr. Tannenbaum's departure. It is a great pleasure to know and have worked with you. Dr. Yezzi's LCCV lab has been my new home since the second-to-last MINERVA member graduated. Thank you, Navdeep, for your devoted work to Vintinura Imaging, which eventually allowed me to work with you and Dr. Yezzi on the STTR project. May your hard work and dedication pay off well in your future Ph.D. studies. Thank you, Samuel, for providing an introduction to adjoint active contours in 2011 and giving me advice on this thesis. In this context, I would also like to thank our "phantom" lab member Dan, with whom I shared some very enlightening and interesting discussions. Also, Ping-Chang, Hong-Ro, and Alireza, it was nice to share the final months of my Ph.D. with you all.

Some of the proof-reading of this thesis was done by Christoph, who not only is a very strict proof-reader, but also a great friend since we met in 2004 as freshmen. Thank you for all the shared memories, both academically and privately. There are many other friends and people, who made these past years very worthwhile and pleasant for me, but who I cannot all name here because it would be impossible to draw an accurate line. In Germany, at Georgia Tech, throughout the US, wherever you are, thank you all for that.

Finally, my parents Eberhard and Ingrid, and my sister Simone deserve special thanks for supporting me throughout my ten years of college education. They are probably the ones who sacrificed the most for this Ph.D. by seeing their son and brother only once or twice per year.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

aMS	anisotropic Mumford-Shah (segmentation energy)
aMS-SR	anisotropic Mumford-Shah with shape-plus-radiance prior
BEM	boundary element method
BVP	boundary value problem
CV	Chan-Vese (segmentation energy)
CV-S	Chan-Vese with shape prior
FDM	finite difference method
FVM	finite volume method
FEM	finite element method
IR	infrared
MFS	method of fundamental solutions
MS	Mumford-Shah (segmentation energy)
NSD	non-smooth data (optical flow)
ODE	ordinary differential equation
OMT	optimal mass transport (optical flow)
PCA	principal component analysis
PDE	partial differential equation
SIFT	scale-invariant feature transform

SUMMARY

In this dissertation, novel variational optical-flow and active-contour methods are investigated to address challenging problems in computer vision and shape-based imaging. Starting from traditional applications of these methods in computer vision, such as object segmentation, tracking, and detection, this research subsequently applies similar active contour techniques to the realm of shape-based imaging, which is an image reconstruction technique estimating object shapes directly from physical wave measurements.

In particular, the first and second part of this thesis deal with the following two physically inspired computer vision applications.

Optical Flow for Vision-Based Flame Detection: Fire motion is estimated using optimal mass transport optical flow, whose motion model is inspired by the physical law of mass conservation, a governing equation for fire dynamics. The estimated motion fields are used to first detect candidate regions characterized by high motion activity, which are then tracked over time using active contours. To classify candidate regions, a neural net is trained on a set of novel motion features, which are extracted from optical flow fields of candidate regions.

Coupled Photo-Geometric Object Features: Active contour models for segmentation in thermal videos are presented, which generalize the well-known Mumford-Shah functional. The diffusive nature of heat processes in thermal imagery motivates the use of Mumford-Shah-type smooth approximations for the image radiance. Mumford-Shah's isotropic smoothness constraint is generalized to anisotropic diffusion in this dissertation, where the image gradient is decomposed into components

parallel and perpendicular to level set curves describing the object's boundary contour. In a limiting case, this anisotropic Mumford-Shah segmentation energy yields a one-dimensional "photo-geometric" representation of an object which is invariant to translation, rotation and scale. These properties allow the photo-geometric object representation to be efficiently used as a radiance feature; a recognition-segmentation active contour energy, whose shape and radiance follow a training model obtained by principal component analysis of a training set's shape and radiance features, is finally applied to tracking problems in thermal imagery.

The third part of this thesis investigates a physics-driven active contour approach for shape-based imaging.

Adjoint Active Contours for Shape-Based Imaging: The goal of this research is to estimate both location and shape of buried objects from surface measurements of waves scattered from the object. These objects' shapes are described by active contours: A misfit energy quantifying the discrepancy between measured and simulated wave amplitudes is minimized with respect to object shape using the adjoint state method. The minimizing active contour evolution requires numerical forward scattering solutions, which are obtained by way of the method of fundamental solutions, a meshfree collocation method. In combination with active contours being implemented as level sets, one obtains a completely meshfree algorithm; a considerable advantage over previous work in this field. With future applications in medical and geophysical imaging in mind, the method is formulated for acoustic and elastodynamic wave processes in the frequency domain.

CHAPTER I

INTRODUCTION

This doctoral research is divided into three parts (Chapters 2, 3 and 4), which represent distinct and self-contained research projects. They are assembled in this thesis under a common theme suggested by the title of this dissertation: All three projects employ variational methods, in particular optical flow (Chapter 2 only) and active contours (Chapters 2, 3, and 4), to approach their respective problems. Moreover, the proposed variational methods are either *motivated* by the physical properties of the problem (Chapters 2 and 3) or even *formulated* based on the physics of the problem (Chapter 4). Our task is to present three very different problems that use very similar techniques.

Each chapter is, therefore, organized as a "thesis within the thesis" providing its own introduction, motivation, literature review, definitions, derivations, results and conclusions. What remains to accomplish in this overall introductory chapter is to put the reader in the position to successfully navigate through this document and to provide basic introductions to the concepts shared by all chapters. These concepts are variational methods in general and active contours in particular (Section 1.2), as well as the level set method for implementing active contour evolution (Section 1.3). First, however, the contents of the three following chapters are briefly summarized.

1.1 Overview

In part one of this research (Chapter 2), the overall goal is to use motion estimation techniques to detect flames in surveillance videos. Active contours are employed to define and track candidate regions, which represent image regions with high probability of fire. Then, four optical flow features are extracted from each candidate

region to characterize the motion field's magnitude and directionality. The features extracted from ground truth data are used to train a neural net that classifies candidate regions as "fire" or "not fire" in the testing phase. Apart from presenting a way to detect flames via optical flow estimation, the contribution of this research is the development of two novel optical flow estimators, which model motion according to the physics of flame dynamics.

In part two (Chapter 3), a series of active contour models for object localization, tracking, detection, and recognition in IR/thermal videos are presented. These models incorporate smooth radiance approximations in the spirit of the classical Mumford-Shah [103] functional, which are motivated by the fact that IR imagery is inherently smoother than visible-spectrum imagery. In terms of mathematical modeling, the heat equation describes thermal distributions (sensed by IR/thermal cameras) as well as the smooth radiance approximation derived from the Mumford-Shah functional. In this project, the anisotropic Mumford-Shah (aMS) energy is formulated as a generalization of the standard Mumford-Shah functional by locally decomposing the gradient operator depending on the geometry of the level set function at a point, and then weighting each gradient component differently in the segmentation energy's smoothness term. In a limiting case, this formulation gives rise to a new and efficient one-dimensional radiance model, which is invariant to translation, scale, and rotation, and which incorporates photometric as well as geometric information. This coupling of radiance and shape is exploited to formulate a recognition-segmentation active contour that restricts the boundary contour and radiance to follow a training model obtained through principal-component-analysis of the training data. These shape-plus-radiance priors are used to increase the robustness of active contour segmentation compared to not using priors or using shape-only priors.

Finally, whereas parts one and two are physically *inspired* computer vision algorithms, part three of this research (Chapter 4) deals with active contours models,

which are *driven* by measurements of physical wave processes, such as acoustic and elastodynamic waves. The goal is to estimate both location and shape of buried objects from surface measurements of waves that scatter from the object. By modeling shapes as active contours, which are evolved to minimize the misfit between measured and simulated wave amplitudes and phases at the receivers, the algorithm directly segments objects circumventing the need for the fine-scale reconstruction of pixel values. Implementing the contours as level sets and employing collocation methods as numerical solvers for the physical boundary value problems (forward problem) leads to a completely meshfree algorithm.

In the remainder of this introduction, variational methods are introduced in Section 1.2 by way of classical examples for image smoothing and image segmentation via active contours. This section is considered a basic, general, and mathematical tutorial on variational methods for the reader to understand the mathematical approaches being taken in this thesis. Active contours are implemented in level set formulation, an introduction of which is given in Section 1.3.

1.2 Variational Methods

The key to understanding variational methods is *calculus of variations*. In fact, one could say that a variational method is nothing else than "applied calculus of variations". Let us explain what this statement means: The mathematical branch of calculus of variations (or functional analysis) occupies itself with optimizing (minimizing or maximizing) functionals, where a functional is a mapping from a function space to the real numbers. In other words, given a functional, which assigns a real number to each input function, calculus of variations provides a methodology to find the function that results in the lowest (in case of minimization) or highest (in case of maximization) output value of the functional. Variational methods apply calculus of variations to a functional that was designed for or is relevant to a particular problem

in engineering or science. In science, this functional is usually deduced from or postulated as a physical principle, whereas in engineering, the functional is designed by the engineer to achieve a certain task. It is the latter that we are concerned with in the case of computer vision and image processing.

This idea is now illustrated by means of a simple and intuitive example, that—even though arguably is not the most useful application of variational methods since filters can achieve the same task much more efficiently—conveys the basic concept of variational methods. Let a gray-scale image $I : \Omega \mapsto \mathbb{R}$ be given which is defined on the image domain Ω. We would like to smooth this image to a certain degree for noise removal, i.e., we seek another image $\tilde{I} : \Omega \mapsto \mathbb{R}$ that is similar to I, but smoother. We can combine these two goals by considering the functional

$$E(f) = \int_{\Omega} (1-\alpha)(I(x) - f(x))^2 + \alpha \|\nabla f(x)\|^2 \, dx, \tag{1}$$

where α is a scalar with $0 \leq \alpha \leq 1$, f is an image $f : \Omega \mapsto \mathbb{R}$, and ∇ is the gradient operator. That is, for a given image I and constant α, E is a functional that maps an input image f to a non-negative real value, i.e., $E : H \mapsto \mathbb{R}_0^+$, where H is the space of real-valued functions (images) defined on Ω. The interpretation of Eq. (1) is fairly intuitive: The integral over Ω essentially means to sum up the integrand for all pixels in the image, and we obtain high values if either I is dissimilar to f or if the gradient of f is large (i.e., if f is not smooth). Thus, if we can find an image $\tilde{I}$ that results in the minimum value for E, i.e.,

$$\tilde{I} = \operatorname*{argmin}_{f \in H} E(f), \tag{2}$$

it is fair to assume that this image will be a good trade-off between our goals of similarity to I and smoothness. Furthermore, we can choose α to emphasize one or the other goal. For example, a value close to $\alpha = 1$ will require the optimal image to be very smooth as almost all the weight is put on the second term. On the other hand, a value close to $\alpha = 0$ will not penalize the gradient of f, and so the optimal

image $\tilde{I}$ will be very similar to I with almost no noise being removed. The design choice to be made by the engineer is, therefore, to find a value for α that results in an acceptable compromise between similarity to the original image and smoothness for noise removal.

From now on, we switch to the more traditional nomenclature of calling a functional an *energy*, as the functionals we define in practical applications usually involve squared terms, which resemble energy formulas from physics. Equivalent terms are misfit or cost. Moreover, without loss of generality, we assume that we are always concerned with minimizing an energy, since a maximization problem can be turned into a minimization problem by considering the negative energy.

The next step towards obtaining an algorithm for image smoothing is to find the optimal $\tilde{I}$ using calculus of variations. We refrain from providing a general introduction to calculus of variations, which has been done before many times, e.g., [54]. Instead, we present and explain the ideas by actually minimizing Eq. (1). Our approach is analogous to what a high-school calculus class teaches about minimizing a function $g : \mathbb{R} \mapsto \mathbb{R}$: Calculate $g'(x)$, the first derivative of $g(x)$, and set it to zero. Then, solve for x. The first derivative to be zero is a *necessary* condition for a local minimum or maximum, not a *sufficient* condition, meaning that the candidate x could be a saddle point. The same argumentation applies to calculus of variations with the only difference that we need to clarify what the derivative of a functional actually means. Since an energy has an infinite-dimensional domain, we need to specify in which dimension we would like to take the derivative, which leads to the definition of the *directional derivative* of an energy in the direction $h \in H$

$$\frac{\partial E(f)}{\partial h} := \lim_{\epsilon \to 0} \frac{E(f + \epsilon h) - E(f)}{\epsilon}. \tag{3}$$

The directional derivative quantifies how the energy changes if we move an infinitesimal step from f in the direction h. Obviously, there are many different directional derivatives, since H is an infinite-dimensional space. To return to the question of how

these derivatives lead to a necessary condition for a minimum, we require that *all* directional derivatives be zero, i.e., for a minimizing function f^* we have the first-order necessary condition that

$$\frac{\partial E(f^*)}{\partial h} = 0 \quad \forall h \in H. \tag{4}$$

In certain cases, this condition can be evaluated, for example, by using Taylor expansions. However, we will take a different approach, which is more relevant to the work in this thesis. The key idea is to introduce an artificial time t and to consider a series of functions $F : \mathbb{R} \mapsto H$ that assigns a function in H to each time t. By substituting $F(t)$ into E, the energy becomes merely dependent on time, i.e., $E = E(t)$, and we can write the derivative with respect to t as

$$\begin{aligned}
\frac{dE(F(t))}{dt} &= \frac{d}{dt}\int_\Omega (1-\alpha)(I - F(t))^2 + \alpha\|\nabla F(t)\|^2\, dx \\
&= \int_\Omega (1-\alpha)2(I - F(t))(-\frac{dF(t)}{dt}) + 2\alpha\, \Delta F(t)\, \frac{dF(t)}{dt}\, dx \\
&= 2\int_\Omega \Big(\alpha\Delta F(t) - (1-\alpha)\big(I - F(t)\big)\Big)\frac{dF(t)}{dt}\, dx,
\end{aligned} \tag{5}$$

where Δ is the Laplacian, i.e., $\partial^2/\partial x^2 + \partial^2/\partial y^2$ in two dimensions, and we can take the derivative inside the integral because the integration domain Ω does not depend on t. Our goal now is to find the series of $F(t)$ that guarantees $\frac{dE(F(t))}{dt}$ to be negative, as this will imply that E decreases over time. It is not hard to show that this requirement leads to the choice of

$$\frac{dF(t)}{dt} = c\, \big(-\alpha\Delta F(t) + (1-\alpha)\big(I - F(t)\big)\big) \tag{6}$$

for some positive non-infinite constant $c > 0$ (which we may choose to be $c = 1$ without loss of generality, since t is an artificial time which may be scaled by c), and Eq. (5) becomes

$$\frac{dE(F(t))}{dt} = -2\int_\Omega \Big(\alpha\Delta F(t) - (1-\alpha)\big(I - F(t)\big)\Big)^2\, dx \leq 0 \tag{7}$$

and equal to zero only if

$$\alpha\Delta F(t) - (1-\alpha)\big(I - F(t)\big) = 0 \quad \forall x \in \Omega. \tag{8}$$

In fact, this last condition is the first-order necessary condition from Eq. (4). This statement is justified by the relationship

$$\begin{aligned}\frac{dE(F(t))}{dt} &= \lim_{\epsilon\to 0}\frac{E(F(t+\epsilon))-E(F(t))}{\epsilon}\\ &= \lim_{\epsilon\to 0}\frac{E(F(t)+\epsilon\dot{F}(t))-E(F(t))}{\epsilon} = \frac{\partial E(F(t))}{\partial \dot{F}(t)},\end{aligned} \tag{9}$$

where we denote $\dot{F}(t) = \frac{dF(t)}{dt}$. The second equivalence can be obtained by Taylor expansions and the last line is according to Eq. (3). Combining Eq. (4), Eq. (5), Eq. (9) it is not hard to show that Eq. (8) must hold.

In summary, Eq. (6) is a time-evolution equation, more specifically a partial differential equation (PDE), that evolves an arbitrary initial image $F(0)$ to a local (if not global) energy minimum F^* as $t \to \infty$. This approach is commonly referred to as *gradient descent*, because the right hand side of Eq. (6) can be interpreted as the negative of the gradient of E with respect to $F(t)$.

In reality, images are discrete pixel images (not functions of continuous variables), so that—in order to implement the gradient descent flow equation Eq. (6)—we need to discretize the equations in space and time and perform the update equations

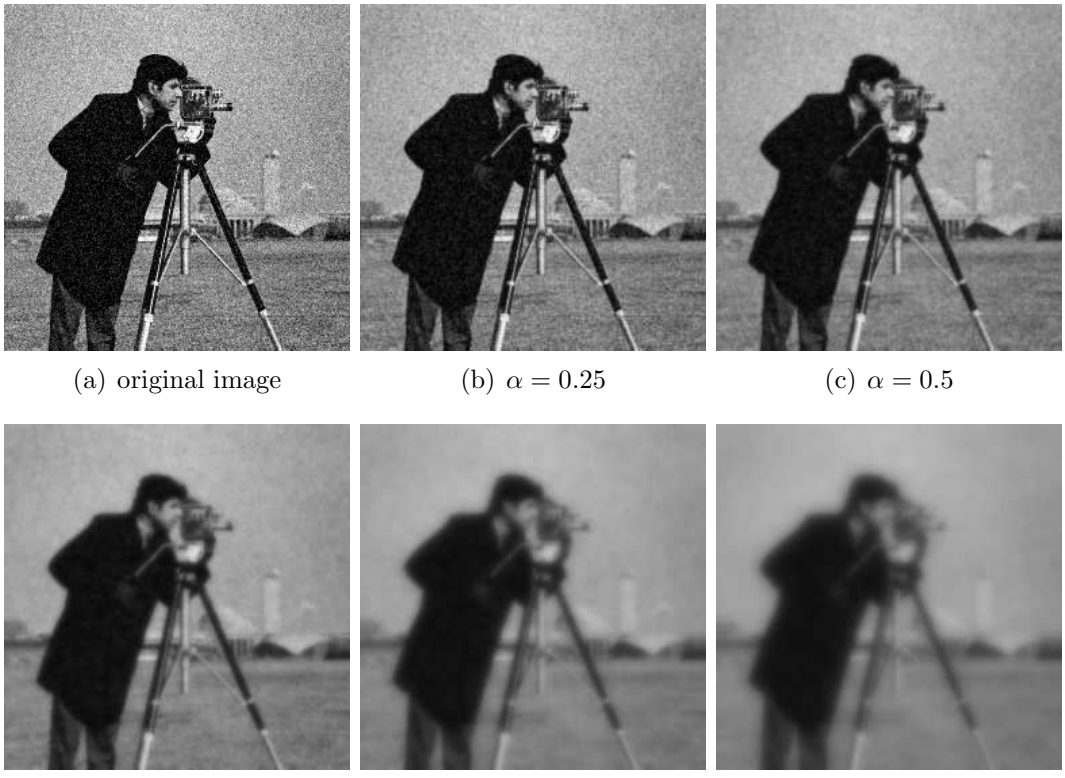
$$F_{i+1} = F_i + \delta t\,(\alpha\Delta F_i - (1-\alpha)(I - F_i)) \tag{10}$$

at each pixel until convergence. Here, $\delta t > 0$ is the time step, F_i is the image at discrete time $t = t_i = i\,\delta t$, $i = 0, 1, 2, ...$, and Δ is a discrete approximation of the Laplacian using central differences. The problem is now in the domain of discrete numerical solutions to PDEs, in particular evolution equations. See, for example, [119] for an introduction to numerical approaches using finite differences, where major concerns are accuracy and stability of the numerical approaches.

The image smoothing energy Eq. (1) is illustrated in Figure 1 by means of the cameraman image (courtesy by Massachusetts Institute of Technology) with added zero mean Gaussian noise. The final converged minimizers of Eq. (1) are shown for

different values of α. As expected, the higher α, the more noise is removed, but also the image becomes blurrier and less similar to the original image.

Whereas the image smoothing energy is a good example to introduce variational methods in image processing, this thesis is mostly concerned with energies that take a *curve* (or *surface* for three-dimensional images) as an input. For ease of presentation, we only consider curves embedded in a two-dimensional image domain in the following. As opposed to an image (or simply "function") that is defined on the image domain, a curve is a subset of the image domain that has one less dimension than the image domain itself. It can be parametrized by a continuous mapping $C(p) : [0, 1] \mapsto \Omega$. Often we consider closed curves where $C(0) = C(1)$. More introductory material on curves and differential geometry can be found in [33].



(a) original image (b) $\alpha = 0.25$ (c) $\alpha = 0.5$

(d) $\alpha = 0.75$ (e) $\alpha = 0.9$ (f) $\alpha = 0.95$

Figure 1: Image smoothing results for the noisy cameraman image by minimizing Eq. (1) for various values of the smoothness weight α.

For image segmentation, one associates the region inside the curve with the *foreground* or *object*, and the region outside the curve with the *background*. Again, we prefer to give an example to provide some intuition. Consider the popular Chan-Vese energy [23]

$$E(C) = \int_R (I - u_{in})^2 \, dx + \int_{R^c} (I - u_{out})^2 \, dx + \gamma \int_C 1 \, ds. \tag{11}$$

With reference to Figure 2, R is the object region (i.e., the region within the curve C), and $R^c = \Omega \setminus R$ is its complement, the background region. We have that $C = \partial R$ is the boundary of the object region. Moreover, u_{in} and u_{out} are the image intensity means in R and R^c, respectively, i.e.,

$$u_{in} = \frac{\int_R I(x) \, dx}{\int_R 1 \, dx}, \quad u_{out} = \frac{\int_{R^c} I(x) \, dx}{\int_{R^c} 1 \, dx}. \tag{12}$$

The last term in Eq. (11) is a regularization term that enforces the optimal curve to be smooth (i.e., zig-zag contours should be avoided) and $\gamma > 0$ is a design parameter determining the degree of curve smoothness regularization. The rationale behind this energy is to assume that object and background have different image intensity means and that one attempts to partition the image such that the mean inside and the mean outside the curve approximately describe this property of the image. Of course, if this assumption about the image is not satisfied, the Chan-Vese energy will perform very poorly. After the energy is defined as above, the next step is again to minimize the energy with respect to C, i.e., to find the curve that results in the minimum energy.

When minimizing with respect to curves, it becomes essential to take the gradient descent evolution approach by letting curves depend on artificial time t. Then, taking the derivative of E with respect to t, it can be shown that the Chan-Vese energy results in the gradient descent flow equations

$$\frac{dC}{dt} = \big((I(x) - u_{out})^2 - (I(x) - u_{in})^2 - \gamma\kappa\big)\mathcal{N}, \tag{13}$$

where κ and $\mathcal{N}$ are the curvature and the outward normal vector of $C(t)$, respectively. The quantities u_{out}, u_{in}, κ and $\mathcal{N}$ depend on the curve and vary, therefore, over time

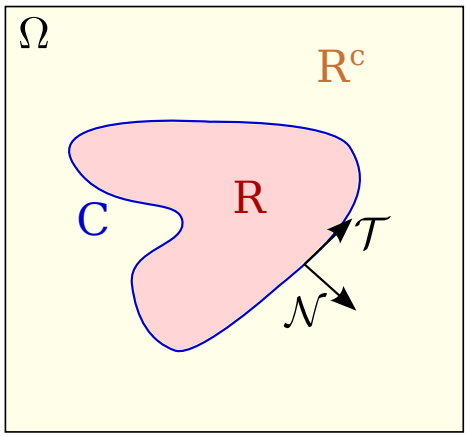


Figure 2: A closed curve C divides the image domain Ω into object region R and background region $R^c = \Omega \setminus R$. If the curve is smooth, orientations for the outward normal $\mathcal{N}$ and the tangential $\mathcal{T}$ are defined as shown.

as well. Moreover, this equation applies point-wise to each point x on the curve, and κ and $\mathcal{N}$ are dependent on x. Because the curve changes over artificial time, the term *active contour* is typically used in computer vision for an energy that depends on a curve and is minimized through this evolution approach.

An example of Chan-Vese curve evolution is shown in Figure 3 with the initial curve being a box in the center of the image. As the gradient descent curve flow progresses, the curve evolves to capture the dark areas of the cameraman, whereas brighter areas belong to the background of this segmentation result.

The Chan-Vese evolution equation Eq. (13) makes sense intuitively as we would like to move the curve outward (inward) where the intensity at a curve point x is better described by u_{in} (u_{out}). Even though one could have devised a flow equation like Eq. (13) from the beginning, one would not have had the insight that this flow is actually the optimal flow when inside and outside intensity is modeled as constant and penalized in the least-squares-sense. In other applications, it might be very difficult to design flow equations directly, whereas design criteria are usually easier to integrate in the energy formulation.

One major advantage of variational methods is their generality in that many types

(a) initial guess (b) 50 iterations (c) 100 iterations

(d) 200 iterations (e) 300 iterations (f) 400 iterations

Figure 3: Chan-Vese segmentation by minimizing Eq. (11) shown at different stages of the gradient descent flow.

of problems, especially non-linear problems, can be approached by these methods, which this thesis will offer a glimpse of. The formulation of an energy is usually straightforward and incorporates the objectives for the solution. Usually, an energy consists of a fidelity term, which penalizes deviations from a model of the problem at hand, and a regularization term, which resolves issues of ill-posedness of the problem and increases robustness by restricting the input space to a smaller set (e.g., the set of "smooth" curves, as opposed to the set of all curves). Once the energy is defined, the remaining task is to minimize the energy using calculus of variations and to implement the resulting minimizing equations. These tasks can be mathematically or computationally complex, but they are straightforward for a seasoned applied mathematician. The resulting algorithm will be optimal in the sense of the energy defined by the user.

Some disadvantages of variational methods (besides their mathematical complexity) were mentioned above. Local minima can pose serious problems depending on the energy. Some energies are very well-behaved and do not have any or very few local minima, such as the the smoothing example shown above. Other applications, such as for example the inverse scattering application presented in Chapter 4, suffer from a high density of local minima. Local minima are undesired because they are not the solution (global optimum) sought after. Calculus of variations provides only necessary conditions for a minimum, and so the minimizing scheme might converge to a local minimum. The density of local minima depends on the ill-posedness of the problem and can be alleviated by adding regularization terms (at the cost of diluting the influence of the fidelity term). Another potential drawback of variational methods is their computational complexity, which might prevent their use in time-critical applications.

1.3 Level Set Method

In general, if an active contour energy depends only on the geometry of the curve C (i.e., the set of points) and not on its parametrization $C(p)$, the energy is called *geometric* and the evolution equation can be written as

$$C_t = F\mathcal{N}, \tag{14}$$

where $C_t = \frac{dC}{dt}$ and $F : C \mapsto \mathbb{R}$ is a scalar force term defined on each point of the curve at time t. It is obvious that at each point on the curve, the change of the curve can be decomposed into a component perpendicular (the normal) and a component parallel (the tangent) to the curve. An infinitesimal change in the tangential component does not alter the geometry of the curve (it alters only its parametrization as points along the curve get redistributed), and thus, the energy does not change because it depends on the geometry of the curve. The tangential component is, therefore, zero and the force term $F\mathcal{N}$ describes the whole curve evolution process.

The most intuitive approach to implementing Eq. (14) for given F, is to represent a curve by a set of ordered marker points, which are updated according to Eq. (14), and the curve is interpolated between the markers. These marker representations are typically referred to as *snakes*, and these implementations were used for active contours in its early stages [76]. There are several drawbacks with this approach, however. For example, one needs to ensure that the marker points remain evenly distributed and that they stay dense, e.g., the number of marker points has to increase as the curve grows. Moreover, handling topological changes such as merging of two curves, dividing a curve into two curves, or cutting out a whole require very special and complicated routines. These drawbacks were addressed by the work of Osher and Sethian, who introduced level set methods for curve (or generally interface) evolution [106]. Next, their idea of representing interfaces implicitly as the zero level set of a higher-dimensional function is presented and motivated.

Level set methods describe time-dependent interfaces (curves in 2D, surfaces in 3D) as the zero level set of a level set function $\psi : \Omega \times [0, \infty) \mapsto \mathbb{R}$, where Ω is the image domain as far as computer vision problems are concerned, so that

$$C = \{(x, t) \in \Omega \times [0, \infty) : \psi(x, t) = 0\}. \tag{15}$$

A level set function (for fixed t) representing the final segmentation of the cameraman in Figure 3 is illustrated in Figure 4. Differentiation of Eq. (15) with respect to t yields

$$\frac{d\psi}{dt} = \psi_t + \nabla\psi \cdot C_t = 0, \tag{16}$$

where ∇ is the spatial gradient. Observing that $\nabla\psi/\|\nabla\psi\| = \mathcal{N}$, we obtain from Eq. (14) and Eq. (16) that

$$\psi_t = -\|\nabla\psi\|\mathcal{N} \cdot F\mathcal{N} = -F\|\nabla\psi\|. \tag{17}$$

This evolution equation for ψ is equivalent to Eq. (14), and the curve C at time t can be extracted as the zero level set of ψ at time t.

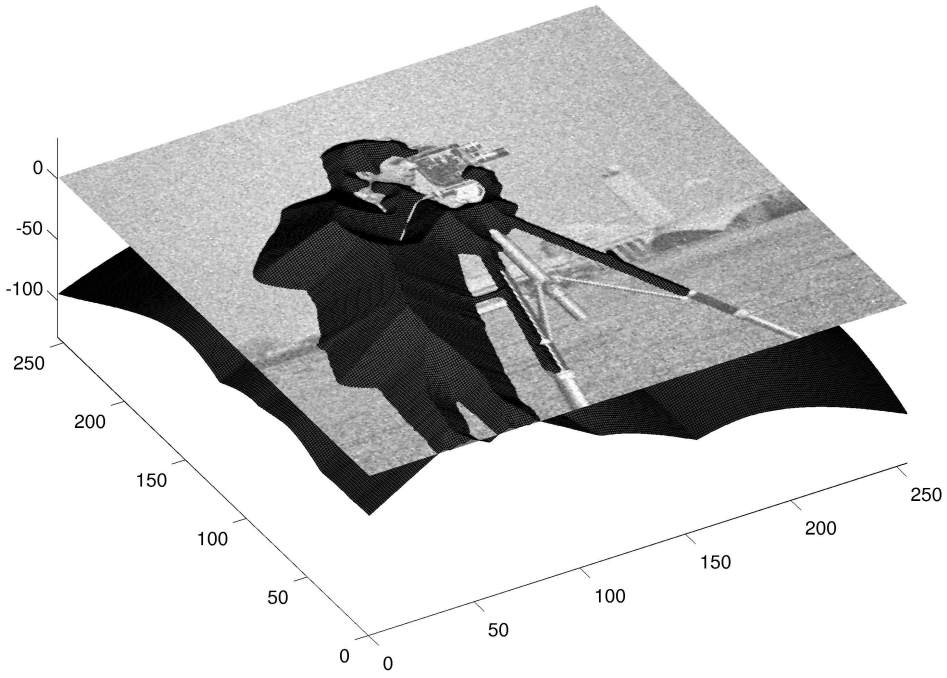


Figure 4: Level set representation for the final segmentation of the camerman from Figure 3.

For terms derived from the fidelity term of an energy, the resulting force F is often an external term that does not directly depend on the geometric properties of the curve at each point. However, for the curve smoothness regularization term such as the last term in Eq. (11), we have that $F = -\gamma\kappa$, i.e., the force is determined by the curvature at a curve point. This flow is called *curvature flow*. Since now F itself is an intrinsic geometric property of the curve, the level set formulation of this regularization flow can be further simplified. Note that the curvature can be computed from the level set function as

$$\kappa = \operatorname{div}\left(\frac{\nabla\psi}{\|\nabla\psi\|}\right) \tag{18}$$

and so the regularization curvature flow becomes

$$\psi_t = \gamma\|\nabla\psi\| \operatorname{div}\left(\frac{\nabla\psi}{\|\nabla\psi\|}\right). \tag{19}$$

In two dimensions with coordinate axes x and y, this expression can be written as

$$\psi_t = \gamma \frac{\psi_{yy}\psi_x^2 - 2\psi_x\psi_y\psi_{xy} + \psi_{xx}\psi_y^2}{\psi_x^2 + \psi_y^2}, \tag{20}$$

where subscripts denote partial derivatives.

Turning to the implementation of the above level set evolution equations, we will also be able to answer why level sets might be advantageous over snakes. Observe that the level set function ψ is defined on the fixed image domain Ω, so that we can essentially treat the level set function like an image, and supposedly we already have the data structure needed to represent and manipulate an image, namely, a pixel grid with stored values and some functions working on images. In order to evolve the level set over artificial time according to Eq. (17) and Eq. (20), the spatial derivatives are discretized and the problem becomes very similar to the smoothing example in Section 1.2, where a function (or image) is evolved on a fixed grid according to a partial differential equation.

The difficulty for the level set method lies in ensuring that the resulting level set evolution is numerically stable. When the force is an external function, Eq. (17) is a transport-type equation with spatially varying speed function F. Spatial derivatives are then discretized using an entropy-conserving upwind scheme that ensures that information is propagated in the correct direction. See [114] for more details. The regularization flow Eq. (20) is also known as the geometric heat equation and, thus, has very similar similar properties to the heat equation. Spatial derivatives are discretized using central differences. Finally, to evolve the level set function, the time derivative is discretized by a forward difference, and the discrete update equation becomes

$$\psi_{i+1} = \psi_i - \delta t \, F_i \, \|\nabla \psi_i\|, \tag{21}$$

where δt is the step size and i is the discrete time parameter ($t = \delta t \, i$). For both flows (external force flow and curvature flow), there exist theoretical bounds on the discrete

time step to ensure stability. In particular, for an external force F, $\delta t \leq 1/\max_C\{|F|\}$ where $\max_C\{|F|\}$ is the maximum absolute value of F along C. For the curvature flow, stability requires $\delta t \leq 1/(2\gamma)$.

Besides the already mentioned advantage of level sets that they use the same data structure as the underlying image, i.e., a pixel grid, level set methods also handle topological changes internally without any additional effort, as values continuously switch sign and thus switch membership between the inside and the outside of the curve. In this context, one might wonder how accurate a level set can actually describe the location of the curve, i.e., if it is possible to obtain sub-pixel resolution. The answer is that sub-pixel accurate locations of curve points can be obtained by interpolating between level set values on the grid and determine the zero level set crossing on this interpolated level set function, so that there is no loss of accuracy due to the pixel grid. Another advantage of level sets over snakes lies in the fact that inside and outside regions can be easily retrieved by checking the sign of the level set function. Without loss of generality, our assumption is that the level set function is negative inside the curve ("object" region) and positive outside ("background" region).

On the negative side, level set evolution can be relatively slow compared to updating several marker points for snakes. This drawback, however, is addressed by narrow-band methods [114] for level sets, where only a narrow neighborhood around the zero level set is actually updated. In return, the increase of speed comes at the cost of more complex implementation and less automatism in topological changes as the contour will be "blind" outside this narrow neighborhood of the curve. In the context of narrow-band methods, fast marching methods [114] play an important role as well. Fast marching methods allow for an efficient coverage of the inside and outside regions, e.g., to build a level set function on the whole grid from a narrow-band level set function. Fast marching methods traverse through the grid on an optimal

path, which is determined by the distance to the curve. Points closer to the curve will be processed first, which makes it possible to construct a connection tree for each point in the domain containing information about the distance to the curve and the closest point on the curve.

The fact that level sets are meshfree, in other words marker-free or element-free, may be viewed as an advantage or disadvantage depending on the application. Generally, many operations on the interface such as integration or interpolation can be performed in the level set domain. Level set representations are a less convenient choice if the connectivity between sample points on the interface is required at some point of the application, since level sets provide only an unsorted set of points comprising the curve or surface.

Finally, a general note is made that there is some confusion in the community between the terms *active contours* and *snakes*. Snakes obviously refer to curve energies, whose flow equations are implemented by marker methods. The term active contour, on the other hand, can be ambiguous due to the following two possible meanings:

1. a curve energy that is minimized via the gradient descent flow equation (independent of how the flow equation is implemented). Thus, snakes are a particular implementation of active contours.
2. a curve energy that is minimized via the gradient descent flow equation, which is then implemented as a level set evolution. Thus, active contours are the counter-part to snakes on the implementation level.

We prefer the second interpretation because we deal with level set implementations exclusively. This choice makes it clear that our work, especially Chapter 4, addresses in particular level set implementations and the application-specific challenges that they entail.

CHAPTER II

OPTICAL FLOW FOR VISION-BASED FLAME DETECTION

2.1 Introduction

Detecting the break-out of a fire rapidly is vital for prevention of material damage and human casualties. This is a particularly serious problem in situations of congested automobile traffic, naval vessels, and heavy industry. Traditional point-sensors detect heat or smoke particles and are quite successful for indoor fire detection. However, they cannot be applied in large open spaces, such as hangars, ships, or in forests. This chapter presents a video-detection approach geared toward these scenarios where point-sensors may fail. In addition to covering a wide viewing range, video cameras capture data from which additional information can be extracted; for example, the precise location, extent, and rate of growth. Surveillance cameras have recently become pervasive, installed by governments and businesses for applications like license-plate recognition and robbery deterrence. Reliable vision-based fire detection can feasibly take advantage of the existing infrastructure and significantly contribute to public safety with little additional cost.

2.1.1 Review

Computer vision concepts are often inspired by human vision. A comprehensive and elegant description of the human perception of fire was presented by the 16th century French poet Du Bartas [40]: "Bright-flaming, heat-full fire, the source of motion." Whereas Du Bartas missed the characteristic reddish color that almost any vision-based detection algorithm builds upon, e.g., [22], his quote covers most of the other

features employed by previous methods. High brightness or luminance causes image pixels to saturate—one feature utilized in [14, 67, 96]. The instantaneous flame-like texture or spatial patterns are used in [67, 91, 131]. Features related to the flame's shape and change of shape are found in [14, 91, 131]. Also, the flickering, typical of fire, presents a popular feature [14, 67], which has also been analyzed in the wavelet domain [120, 86]. This work is restricted to studying images in the visible spectrum and emitted heat or infrared light is not considered, but it is used in a number of approaches where infrared sensors are available [125]. Lastly, fire motion offers a whole suite of possible features. The authors of [109] consider the temporal variation of pixel intensity for fixed pixel locations. More recently, [61] has developed a more comprehensive statistical approach by computing covariances of image quantities on small blocks, thus making use of spatial and temporal patterns. These statistics are applied directly to the image data and these methods are therefore comparably efficient. Optical flow estimators, on the other hand, transform the image sequence into estimated motion fields, allowing for a more insightful extraction of features. Classical optical flow algorithms are analyzed in [46] for the recognition of various dynamic textures.

2.1.2 Overview

This research adds to the existing literature on vision-based flame detection by using optical flow features, which has not been attempted before. The computation of optical flow and feature extraction from optical flow is embedded in a general detection framework, which is presented first in Section 2.2. The framework is based on the tracking of suspicious regions (candidate regions) via active contours. This approach allows for temporal tracking of classification results and suppresses background clutter in the feature extraction process. A general introduction on optical flow and the specific implementations used in this research are shown in Section 2.3, where the key

idea is the use of optimal mass transport (OMT) optical flow that models fire motion according to the mass conservation equation, a physical equation characterizing fluid and gas flow. Moreover, a data-driven, non-smooth optical flow estimator (NSD) is presented for saturated flames. The features extracted from optical flow fields are defined and motivated in Section 2.4. Detection results on a large video database are reported in Section 2.5, and final conclusions for this project are drawn in Section 2.6.

2.2 Detection Framework

The feature extraction and classification scheme is embedded in a track-before-detect framework, which uses active contours to define and track candidate regions. The overall framework is shown in Figure 5: In order to decrease the nuisance factor from background structure, suspicious regions are segmented and labeled as candidate regions. Tracking candidate regions over time allows for more robust results because classification results from several time steps can be incorporated in the decision.

To initialize a candidate region, background subtraction and color thresholding is applied to the image, and pixels with motion and fire-like color are combined into candidate regions by k-means clustering [73]. Each cluster serves as the initialization for an active contour, which further segments the candidate region more accurately.

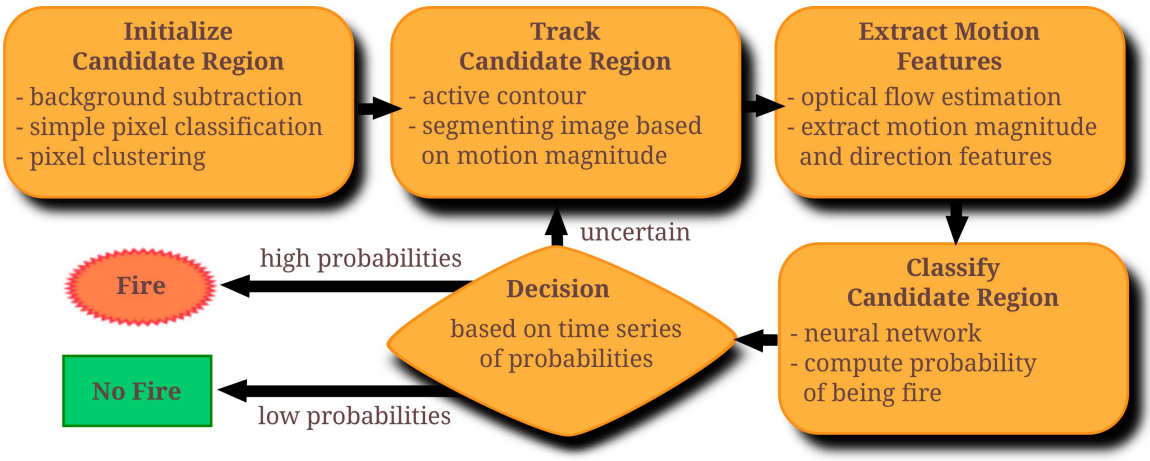


Figure 5: The flame detection framework.

This task is achieved via a motion segmentation technique where contours are driven to separate the mean [23] of the magnitude of the OMT optical flow [87] inside and outside the contour, thus segmenting moving objects from the static background. Candidate regions are then tracked over time by evolving the contour from the previous frame. This simple tracker is sufficient for stationary fire flames and slowly moving background objects.

Within a candidate region, feature extraction and classification is performed. For classification, a neural net [9] is employed, which was trained on hand-labeled ground truth data. In the classification phase, the neural net assigns the probability of fire to a candidate region. Based on the time series of probabilities coming from subsequent frames, candidate regions are deleted (very low time-averaged probability output, class = no fire), being tracked (medium time-averaged probability output, class undecided), marked as fire (high time-averaged probability output, class = fire).

Regarding feature extraction, a pixel-wise approach was used in early results of the project, where features for each pixel were taken to be the color and motion magnitude of the pixel and its neighbors as described in [87]. By thresholding each pixel's classification output, a binary map of fire pixels is obtained, which are counted in relation to the total number of pixels in the candidate region. This ratio of number of fire pixels versus number of pixels within the contour is monitored over time and a fire is detected if the ratio consistently exceeds a certain threshold. Using this pixel-wise classification approach, the active contour tracking framework is illustrated in Figure 6 and Figure 7. In Figure 6, two candidate regions are detected (the actual fire and a fireman), but only the true fire region is eventually classified as fire. These results are all obtained using the same neural net and were presented to the scientific community in [101].

(a) frame 1 (b) frame 3 (c) frame 5 (d) frame 12

Figure 6: Result for a video with a burning van. Blue contour: tracked candidate region. Red contour: detected region. (a) Initial candidate regions, (b) candidate regions are tracked, (c) fire is detected, (d) yellow, moving fireman is classified as 'not fire' and contour is deleted.

(a) tunnel (b) tree (c) outdoor (d) indoor with reflection

Figure 7: Other flame tracking results. Blue contour: tracked candidate region. Red contour: detected region.

Whereas these five examples exhibit perfect detection rates, more thorough testing revealed that improvements on the selected features were needed for the detection algorithm to work in the presence of likely false positives such as bright and orange-colored moving objects. An increase in accuracy was expected from taking a region-based classification approach that takes into account motion directionality and global motion statistics. This new feature extraction scheme based on optical flow is presented in [102], which can be easily combined with the active contour tracking framework presented in this chapter. The improved region-wise feature extraction and classification procedure from [102] is outlined in Figure 8 and will be the focus for the remainder of this chapter.

First, however, Section 2.3 introduces optical flow. Since classical optical flow methods are based on assumptions, e.g., intensity constancy and flow smoothness, which are not met by fire motion, we derive two optical flow estimators, which are specifically designed for the detection of fire; OMT exploits the dynamic texture of

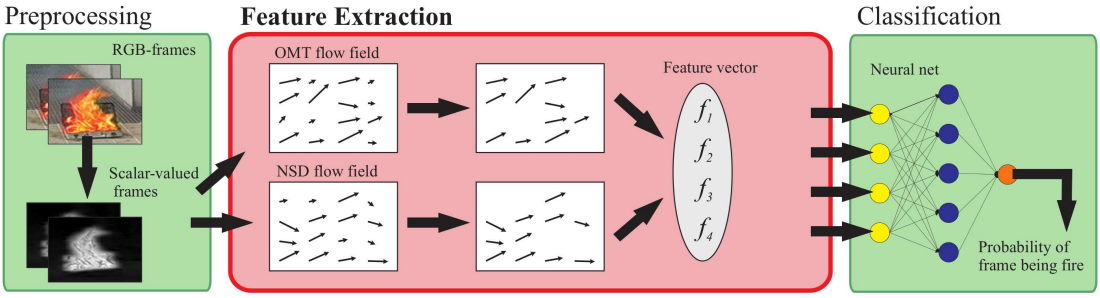


Figure 8: Candidate region classification approach: The main focus of this research is on the extraction of motion features. After preprocessing, two optical flow fields (OMT and NSD) are computed in parallel from which the four-dimensional feature vector is extracted. The feature vector serves as input to a neural net for supervised classification.

flames, whereas a non-smooth modification of classical optical flow, the NSD optical flow, models saturated flames with no dynamic texture. Then, the new set of region-wise optical flow features for fire detection is presented in Section 2.4; these features characterize magnitude and directionality of motion vectors. In Section 2.5, the detection results are illustrated and discussed.

2.3 Optical Flow Estimation

2.3.1 Review

This section provides a general review of optical flow estimation to put the below derived OMT and NSD optical flows into context. Optical flow estimation computes correspondence between pixels in the current and the previous frame of an image sequence. Central to most approaches in establishing this correspondence is the assumption of intensity constancy: moving objects preserve their intensity values from frame to frame. This assumption leads to the optical flow constraint

$$\frac{\mathrm{d}}{\mathrm{d}t} I = \nabla I \cdot \vec{\mathbf{u}} + I_t = I_x u + I_y v + I_t = 0, \tag{22}$$

where $I(x, y, t)$ is a sequence of intensity images with spatial coordinates $(x, y) \in \Omega$ and time variable $t \in [0, T]$, and subscripts denote partial derivatives. The flow vector $\vec{\mathbf{u}} = (u, v)^T = (x_t, y_t)^T$ points into the direction where the pixel (x, y) is moving. In

Eq. (22), I_x, I_y, and I_t are given image quantities and the flow components u and v are unknown. This problem is ill-posed because there are two unknowns and one equation per pixel. This problem is known as the aperture problem, which states that only the optical flow component parallel to the image gradient can be computed.

To obtain a unique solution, optical flow algorithms make further assumptions on the flow field, traditionally by enforcing flow field smoothness. Whereas Lucas-Kanade optical flow [94] is an early representative of methods that assume flow constancy for pixels in a neighborhood, our work follows the point-wise approach, which applies conditions per pixel instead of constant neighborhoods. Point-wise methods generally attempt to minimize a functional of the form

$$E(\vec{\mathbf{u}}) = \int_\Omega \int_0^T r_{\text{data}}(I, u, v) + \alpha r_{\text{reg}}(u, v)\ dt\, dx\, dy, \tag{23}$$

where the data (fidelity) term r_{data} quantifies the deviation from the optical flow constraint Eq. (22) and the regularization term r_{reg} determines the smoothness of the flow field. The constant α controls regularization. In the pivotal paper by Horn-Schunck [68], the data and regularization terms are chosen as

$$E_{\text{HS}}(\vec{\mathbf{u}}) = \int_\Omega \int_0^T (I_t + I_x u + I_y v)^2 + \alpha(\|\nabla u\|_2^2 + \|\nabla v\|_2^2)\ dt\, dx\, dy. \tag{24}$$

From this point, numerous advances have been achieved mostly by changing the regularization term to be image-driven or anisotropic. The introductory sections of [2] provide a comprehensive survey of the developments since. The original optical flow constraint remains most commonly used, either in the original L^2-form (minimizing the L^2-norm of the optical flow constraint) or in other forms such as L^1- or total variation minimization [126]. But also photometrically invariant features to increase robustness to illumination changes, such as, for example, image gradient constraints [107], and appearance models [113] have been incorporated in the data term. Plenty of options have also been studied for the regularization term. The classical L^2-minimization of the flow gradient has been extended to more robust measures,

such as total variation [10]. Also, spatially weighted penalty terms and anisotropic smoothness constraints (to allow for flow discontinuity at object boundaries) have been proposed [127].

Traditionally, optical flow algorithms model rigid object motion, since this type of motion is most often observed in videos or provides a sufficiently accurate approximation for the actual motion. Yet, there are exceptions, for example: Some dynamic textures, e.g., a stationary flame, expose no rigid motion at all, but rather distinct spatio-temporal patterns. For some dynamic textures the optical flow constraint may still be accurate. For example, [46] analyzes how well standard optical flow can distinguish between different dynamic textures. With the introduction of the brightness *conservation* constraint [7], a more general motion model inspired by fluid dynamics has been proposed. This model does not require a pixel to flow with constant brightness, but allows brightness to diffuse to neighboring pixels as well. This more general model is used in [45] for motion segmentation of dynamic textures. An important relationship between the brightness conservation constraint and the Monge-Kantorovich [75] problem (which is a different name for OMT) is shown in [5]. In terms of image processing, the OMT seeks the correspondence map between two images, which minimizes the mass (that is brightness in imaging nomenclature) transportation energy. The OMT problem has been extensively studied mathematically [52] and applied to image registration [63]. Existing OMT algorithms [60], however, are computationally expensive, and therefore, [87] proposes the OMT optical flow, which combines the OMT transport energy with brightness conservation as a relaxed constraint. The OMT optical flow from [87] is revisited in the following section.

2.3.2 Optimal Mass Transport Optical Flow

Classical optical flow models based on brightness constancy, $\frac{\mathrm{d}}{\mathrm{d}t}I = 0$, are inadequate to model the appearance of fire for two reasons. First, fire does not satisfy the intensity constancy assumption Eq. (22), since rapid (both spatial and temporal) change of intensity occurs in the burning process due to fast pressure and heat dynamics. Second, smoothness regularization may be counter-productive to the estimation of fire motion, which is expected to have a turbulent, i.e., non-smooth, motion field. For these reasons, an optical flow estimation modeling fire as a dynamic texture, the OMT optical flow, was introduced in [87].

The optical flow problem is posed as a generalized mass—representing image intensity I—transport problem, where the data term enforces mass conservation. The conservation law is written as

$$I_t + \nabla \cdot (\vec{\mathbf{u}} I) = 0, \tag{25}$$

where $\vec{\mathbf{u}} = (u, v)^T$. With intensity I replaced by mass density, Eq. (25) is known in continuum mechanics [95] as the *continuity equation*, which together with conservation of momentum and conservation of energy form the equations of motion for inviscid fluids (liquids and gases), such as fire. Therefore, Eq. (25) models the data term after a physical law that fire must obey. Further motivation for the use of intensity conservation for dynamic textures can be found in [45].

Analogous to standard optical flow, the OMT optical flow model minimizes the total energy

$$E_{\mathrm{OMT}}(\vec{\mathbf{u}}) = \frac{1}{2} \int_{\Omega} \int_{0}^{T} \left(I_t + \nabla \cdot (I\vec{\mathbf{u}})\right)^2 + \alpha \, \|\vec{\mathbf{u}}\|_2^2 I \, dt \, dx \, dy \tag{26}$$

subject to the boundary conditions $I(x, y, 0) = I_0(x, y)$ and $I(x, y, 1) = I_1(x, y)$, where I_0 and I_1 are given gray-scale images. The transport energy $\|\vec{\mathbf{u}}\|_2^2 I$, which is the "work" needed to move mass from a location at $t = 0$ to another location at $t = 1$,

plays the role of the regularization term. A similar formulation of the OMT problem with mass conservation Eq. (25) being a strict constraint, was introduced by [5].

The solution to this minimization problem is obtained through a "discretize-then-optimize" approach: Start by discretizing Eq. (26) as

$$\frac{\alpha}{2}(\vec{u}^T\widehat{I}\vec{u}) + \frac{1}{2}(I_t + \begin{bmatrix} D_xI & D_yI \end{bmatrix}\vec{u})^T(I_t + \begin{bmatrix} D_xI & D_yI \end{bmatrix}\vec{u}), \tag{27}$$

where $\vec{u}$ is a column vector containing u and v on each pixel, and $\widehat{I}$ is a matrix containing the average intensity values $(I_0 + I_1)/2$ on its diagonal. The derivatives are discretized by $I_t = I_1 - I_0$ and the central-difference sparse-matrix derivative operators D_x and D_y. Now, the quantities $b = -I_t$, $A = \begin{bmatrix} D_xI & D_yI \end{bmatrix}$ are defined and it becomes clear that the function to be minimized is quadratic in $\vec{u}$

$$\frac{\alpha}{2}(\vec{u}^T\widehat{I}\vec{u}) + \frac{1}{2}(A\vec{u} - b)^T(A\vec{u} - b). \tag{28}$$

Taking the derivative with respect to $\vec{u}$ and setting it zero, the solution is obtained as

$$\vec{u} = (\alpha\widehat{I} + A^TA)^{-1}(A^Tb). \tag{29}$$

Since all matrices are sparse, the inversion can be performed quickly by numerical solvers.

In the original OMT optical flow [87] formulation, generalized mass is the pixel intensity I. This choice is based on the assumption that high gas density translates to high intensity in the image. However, there are other phenomena that also map to high intensities, such as the sky or a white wall. In order to improve the chance of good segmentation between foreground and background, this section introduces a new model for the generalized mass based on flame color.

Briefly, the generalized mass of a pixel is represented by the similarity to a center fire color in the HSV color space ($H, S, V \in [0, 1]$), which is chosen to be $H_c = 0.083$, $S_c = V_c = 1$, a fully color-saturated and bright orange. Given the H, S, and V values

of a pixel, the generalized mass is then computed as

$$I = f(\min\{|H_c - H|, 1 - |H_c - H|\}) \cdot S \cdot V, \tag{30}$$

where f is the logistic function

$$f(x) = 1 - (1 + \exp(-a \cdot (x - b)))^{-1} \tag{31}$$

and $a = 100$, $b = 0.11$ are chosen to give the weight function in Figure 9. The hue bar in the x-axis of Figure 9 illustrates which colors correspond to different hue values. A larger value of b would include more of the colors green and purple, which are atypical of fire. A smaller value would exclude red and yellow hues. a is the slope of the transition and, from our experience, is not a sensitive parameter, whereas b should be chosen to reasonably capture fire hue. The color value V corresponds to intensity and the color-saturation S plays the role of dismissing camera-saturated pixels. Although pixel saturation (i.e., color-saturation is low) is sometimes used as a feature for fire detection, saturated regions are useless in the optical flow context due to a lack of image gradient and dynamic texture. The transformation in Eq. (30) weights highly only the periphery of saturated fire regions where camera-saturation tends to occur less. This property becomes visible in Figure 10(c) and Figure 10(d), where the core of the fire center is saturated. The fire texture in Figure 10(a), on the other hand, is preserved in the generalized mass Figure 10(b).

Note that fire classification is not performed based on the output of Eq. (30); this transformation is a preprocessing step to obtain the image on which optical flow is

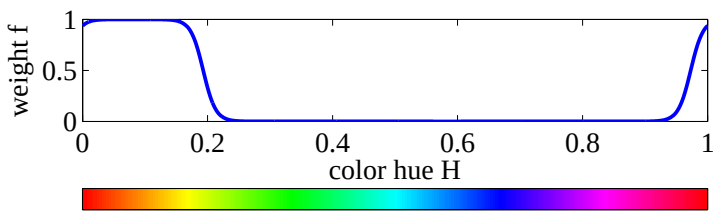
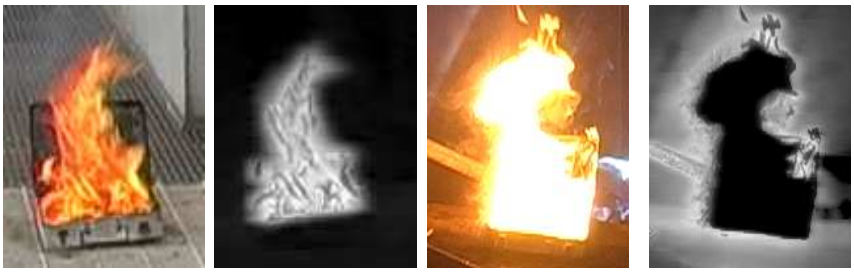


Figure 9: Hue term $f(\min\{|H_c - H|, 1 - |H_c - H|\})$ in Eq. (30).

computed. For this reason, the proposed color model does not have to be as accurate as when used as a feature. Figure 11 shows two examples of OMT flow fields computed from the generalized mass images. It illustrates OMT's ability to capture dynamic texture for the fire image and to discriminate between the rigid object's flow field, which appears much more structured.

2.3.3 Non-Smooth Data Optical Flow

Under unfavorable lighting conditions, especially in closed spaces, fire blobs are likely saturated, thus violating OMT's assumption that dynamic texture is present for fire. Nevertheless, these blobs have boundary motion, which may be characterized by another type of optical flow estimation. An optical flow energy functional called



(a) (b) (c) (d)

Figure 10: Two examples for the generalized mass transformation Eq. (30). (a) and (c): original images. (b) and (d): respective generalized mass (black - 0, white - 1). Fire texture is preserved, saturated regions are assigned low mass.

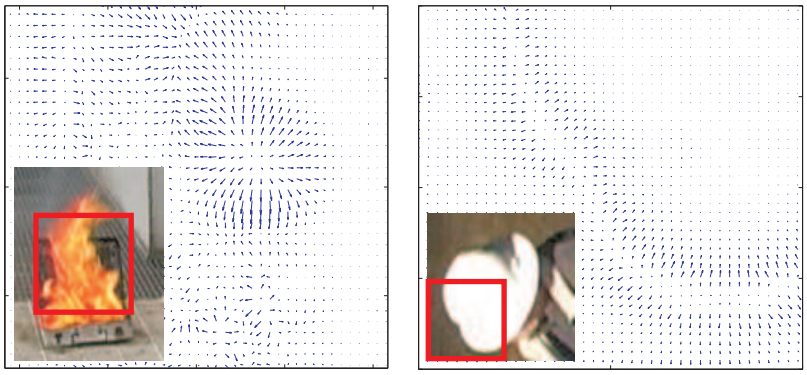


Figure 11: OMT flow fields: fire with dynamic texture (left) and a white hat (right) moving up/right. The red box indicates the area for which the flow field is shown.

Non-Smooth Data (NSD) optical flow and tailored to saturated fire blobs is proposed as

$$E_{\text{NSD}}(\vec{\mathbf{u}}) = \frac{1}{2}\int_{\Omega}\int_{0}^{1} \left(I_t + \nabla I \cdot \vec{\mathbf{u}}\right)^2 + \alpha \left\|\vec{\mathbf{u}}\right\|_2^2 dt\, dx\, dy. \tag{32}$$

The choice of the data term being the optical flow constraint Eq. (22) is justified because pixel saturation trivially implies intensity constancy. Also, NSD is explicitly chosen to be non-smooth since saturated fire blobs are expected to have non-smooth boundary motion. The norm of the flow vector $\|\vec{\mathbf{u}}\|_2^2$ regularizes the flow magnitude, but does not enforce smoothness. This choice makes the NSD flow directions therefore purely driven by the data term under the constraint that flow magnitudes are not too large. While this method is not expected to perform well for standard optical flow applications where flow smoothness plays an important role, it proves useful for detecting saturated fire.

The solution to the minimization problem Eq. (32) is as simple as applying basic arithmetic operations to pre-computed image properties (as opposed to solving a system of equations), which means the NSD optical flow is computationally inexpensive. The Euler-Lagrange equations of Eq. (32) are written as

$$\frac{\partial L}{\partial u} = \left(I_t + \nabla I \cdot \vec{\mathbf{u}}\right) I_x + \alpha u = 0, \tag{33a}$$

$$\frac{\partial L}{\partial v} = \left(I_t + \nabla I \cdot \vec{\mathbf{u}}\right) I_y + \alpha v = 0, \tag{33b}$$

which, after a few manipulations, yield the solution

$$u = -\frac{I_x I_t}{\|\nabla I\|_2^2 + \alpha}, \quad v = -\frac{I_y I_t}{\|\nabla I\|_2^2 + \alpha}. \tag{34}$$

Again, I_x, I_y, and I_t are pre-computed image derivatives and α is the regularization parameter. Figure 12 shows NSD flow fields computed for a saturated fire and a rigidly moving hat. Fire motion on the boundary is non-smooth in contrast to the hat's motion—a result of the data-driven formulation of the NSD.

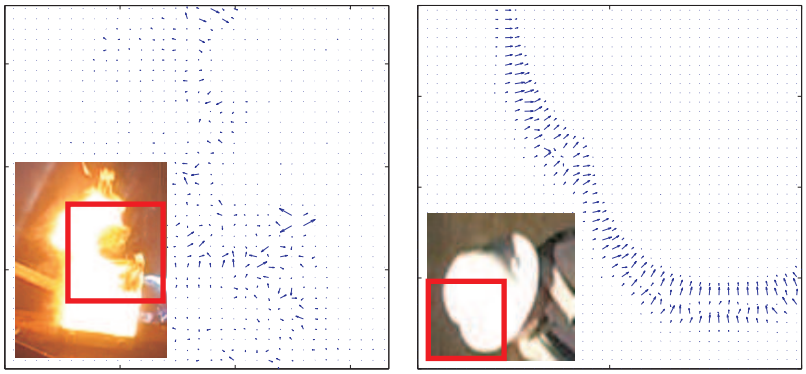


Figure 12: NSD flow fields: saturated fire (left) and a white hat (right) moving up/right. The red box indicates the area for which the flow field is shown.

2.4 Feature Extraction

The computation of optical flow in Section 2.3 is an intermediate step that provides a data set from which motion features can be extracted more intuitively than from the original image. A region-based as opposed to a pixel-based approach is pursued. Whereas the pixel-wise approach classifies each pixel, the region-wise approach aims to classify a region as a whole by analyzing the set of all pixel values (in this case: flow vectors) in that region. In the following section, our approach to feature extraction is defined and motivated.

2.4.1 Pre-Selection of Essential Pixels

Consider a candidate region $\Omega \subset \mathbb{R}^2$. We intend to consider only pixels with relatively high motion magnitude, where motion analysis can be performed reliably. Assume that the optical flow field $\vec{\mathbf{u}} : \Omega \mapsto \mathbb{R}^2$ is computed. Then, the set of essential ("sufficiently moving") pixels $\Omega_e \subset \Omega$ is defined as

$$\Omega_e = \left\{ (x, y) \in \Omega : \|\vec{\mathbf{u}}(x, y)\|_2 > c \cdot \max_{\Omega} \|\vec{\mathbf{u}}\|_2 \right\}, \tag{35}$$

where $0 \leq c < 1$ is chosen such that a sufficiently large number of pixels is retained. In case of extreme outliers, the parameter can be adjusted adaptively. For the test data in Section 2.5, a value of $c = 0.2$ provides sufficient motion segmentation across

the database.

2.4.2 Definition of the Features

In [46] a list of optical flow features is introduced, which is complete in that it considers all possible first-order distortions of a pixel. Those distortions are then averaged within a spatio-temporal block to yield the probability of a characteristic direction or of a characteristic magnitude. [46] also mentions that the highest discriminating power comes from the characteristic direction and magnitude of the flow vector itself, not considering distortions. This observation matches our experience from early prototyping where flow derivatives seemed to have very little discriminating power and were thus removed from further consideration. Given the flow vectors, we propose four features f_i, $i = 1, \cdots, 4$ defining the four dimensional feature vector $F = (f_1, f_2, f_3, f_4)^T$. Given an image region Ω and the optical flow fields $\vec{\mathbf{u}}_{\text{OMT}}$ and $\vec{\mathbf{u}}_{\text{NSD}}$ in that region, the features are chosen as follows.

OMT Transport Energy. This feature measures the mean OMT transport energy per pixel in a subregion

$$f_1 = \underset{\Omega_e}{\text{Mean}} \left(\frac{I}{2} \|\vec{\mathbf{u}}_{\text{OMT}}\|_2^2 \right) . \tag{36}$$

After the color transformation in Section 2.3.2, fire and other moving objects in the fire-colored spectrum, are expected to produce high values for this feature.

NSD Flow Magnitude. Similarly, the mean of the regularization term of the NSD optical flow energy Eq. (32) constitutes the second feature

$$f_2 = \underset{\Omega_e}{\text{Mean}} \left(\frac{1}{2} \|\vec{\mathbf{u}}_{\text{NSD}}\|_2^2 \right) . \tag{37}$$

The first two features, f_1, f_2, will have high values for moving, fire-colored objects. The last two features distinguish turbulent fire motion from rigid motion by comparing flow directionality, as is shown next.

OMT Sink/Source Matching. It is known [52] that solutions to OMT problems are curl-free mappings. For the turbulent motion of fire, this property tends to create flow fields with vector sinks and sources, a typical case of which is displayed in Figure 13(b). For rigid motion, the flow field tends to be comprised of parallel vectors indicating rigid translation of mass.

The third feature is designed to quantify how well an ideal source flow template matches the computed OMT flow field. The template is defined as

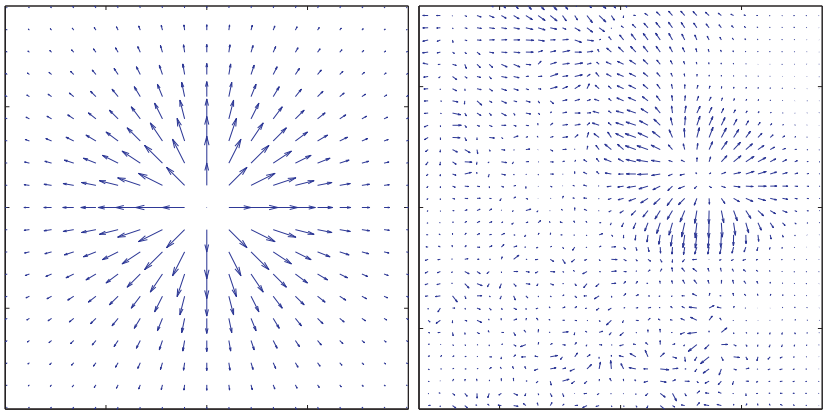
$$\vec{\mathbf{u}}_T(x,y) = \begin{pmatrix} u_T(x,y) \\ v_T(x,y) \end{pmatrix} = \exp\left(-\sqrt{x^2+y^2}\right) \begin{pmatrix} x \\ y \end{pmatrix} \tag{38}$$

and shown in Figure 13(a). Then, the best match is obtained by

$$f_3 = \max_{\Omega} \left| \left(u_T \star \frac{u_{\text{OMT}}}{\|\vec{\mathbf{u}}_{\text{OMT}}\|_2} \right) + \left(v_T \star \frac{v_{\text{OMT}}}{\|\vec{\mathbf{u}}_{\text{OMT}}\|_2} \right) \right|, \tag{39}$$

where $\star$ denotes convolution.

NSD Directional Variance. The final feature distinguishes the boundary motion of saturated fire blobs from rigidly moving objects. This feature analyzes the flow



(a) Ideal source flow template. (b) OMT flow field.

Figure 13: (a) Ideal source flow template and (b) OMT flow field for the fire image in Figure 11. The source matching feature is obtained by the maximum absolute value of the convolution between (a) and (b).

field histogram

$$h(r, \phi) = \underset{\Omega_e}{\text{Hist}}(\vec{\mathbf{u}}_{\text{NSD}}) \tag{40}$$

obtained by kernel density estimation [15]. For notational convenience, the histogram $h(r, \phi)$ is written in polar coordinates, where $r = \sqrt{u^2 + v^2}$ represents the flow vector magnitude and $\phi = \arctan(v/u)$ the flow vector direction. Figure 14 shows the motion histograms of the flows from Figure 12. As expected, the flow vectors for fire motion are scattered around the origin, whereas the flow vectors for rigid motion point predominantly in one direction (up and to the right, in this case). This observation is quantified by the fourth feature as

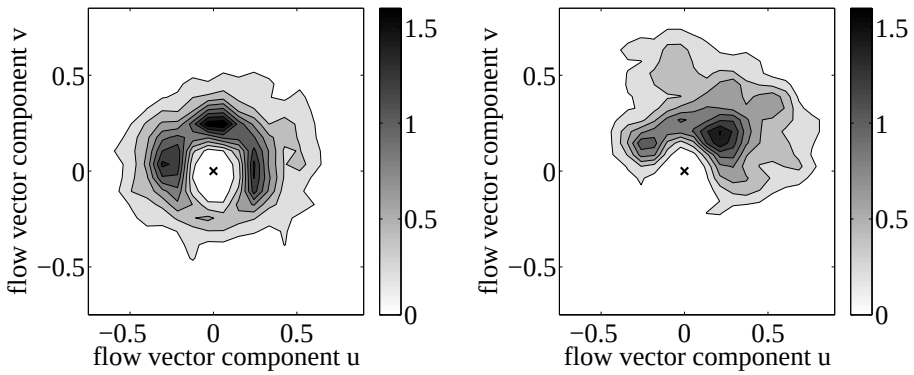
$$f_4 = \text{Var}\{s_i,\, i = 0, ..., n-1\}, \tag{41}$$

where

$$s_i = \frac{\int_0^\infty \int_{2\pi i/n}^{2\pi(i+1)/n} h(r, \phi)\, d\phi\, dr}{\int_0^\infty \int_0^{2\pi} h(r, \phi)\, d\phi\, dr} \tag{42}$$

for some positive integer n. Each s_i represents the ratio of pixels whose flow vectors' angles are between $2\pi i/n < \phi \leq 2\pi(i+1)/n$. For a flow field pointing mainly in one direction, only a few of the s_i are large, and consequently, the value for f_4 is high. On the other hand, if the proportion of vectors pointing in each direction is approximately equal, all of the s_i will be similar, and f_4 will have a low value.

Feature f_4 is more versatile for detecting rigid motion than the probability of a characteristic flow direction in [46]: For example, consider a flow field where half of the pixels point in one direction, and half in the opposite direction, which we would associate with rigid motion having two characteristic directions. Feature f_4 will still yield a fairly high value indicating rigid motion, whereas in [46], the probability of a characteristic motion is practically zero since opposite vectors cancel each other out.



(a) Fire motion histogram.

(b) Rigid motion histogram.

Figure 14: Motion histograms for the flows in Figure 12. (a) (fire) has a multi-directional distribution, whereas (b) (hat) is dominantly moving up and right.

2.5 Results

2.5.1 Detection Rates on Test Database

The algorithm is tested on a database provided by United Technologies Research Center (UTRC), Connecticut, USA. The database features various scenarios including indoor/outdoor, far/close distance, different types of flames (wood, gas, etc.), changing lighting conditions, partial occlusions, etc. The videos have a frame rate of 30 frames/second and spatial dimension of 240 by 360 pixels. From each of the 263 scenarios (containing 169 fire and 94 non-fire sequences), 10 consecutive frames are labeled as ground truth providing a test database of 2630 frames. Note that the non-fire scenarios are chosen to be probable false positives, namely moving and/or fire-colored objects such as cars, people, red leaves, lights and general background clutter. A neural network [9] was trained on frames from 20 of those videos not used in the test database. The test result shows that fire is reliably detected using optical flow features only: the false positive rate (fire is detected when no fire is present) is $f_p = 3.19\%$ and the false negative rate (no fire detected where there is a fire) is $f_n = 3.55\%$. These false detections rates are competitive with recently published, full-fledged fire detection systems, e.g. [22]: $f_p = 31.5\%$, $f_n = 1.0\%$, [14]:

$f_p = 0.68\%$, $f_n = 0.028\%$, [67]: $f_p = 0.30\%$, $f_n = 12.36\%$, [61] (average among test videos): $f_p = 3.34\%$, $f_n = 21.50\%$.

2.5.2 Qualitative Analysis

Example scenarios from the above tests are given in this section to illustrate successful classifications as well as typical false detections and their causes. Figure 15 illustrates successful fire detections and Figure 16 shows two examples of correctly classified non-fire scenes. In Figure 16, the printing on the jacket and the truck entering at the bottom right are moving, fire-like colored (the truck has a reddish tint) objects. Nevertheless, due to their structured motion, these scenarios are correctly classified as not fire.

Occasional false negative detections are observed in four types of scenarios. First, horizontal lines resulting from structured noise in Figure 17(a) introduce structure to the otherwise turbulent fire motion. Therefore, structured noise must to be avoided when using the proposed features. Second, in Figure 17(b), the flame color is oddly

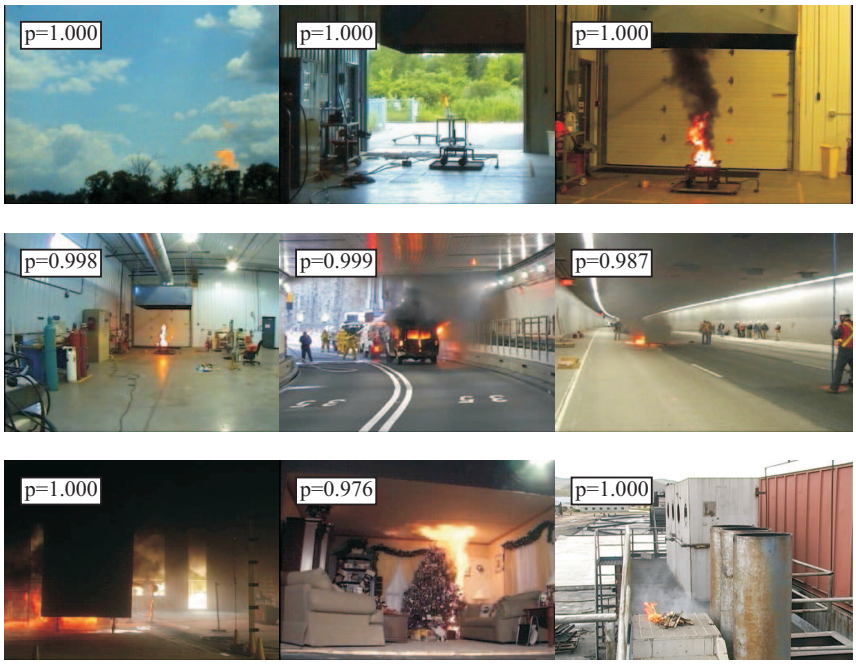


Figure 15: Examples of detected fire scenes with the resulting probabilities.

(a) fire-colored moving text (b) truck entering (bottom right)

Figure 16: Examples of potential false positives correctly classified as non-fire scenes with the resulting probabilities.

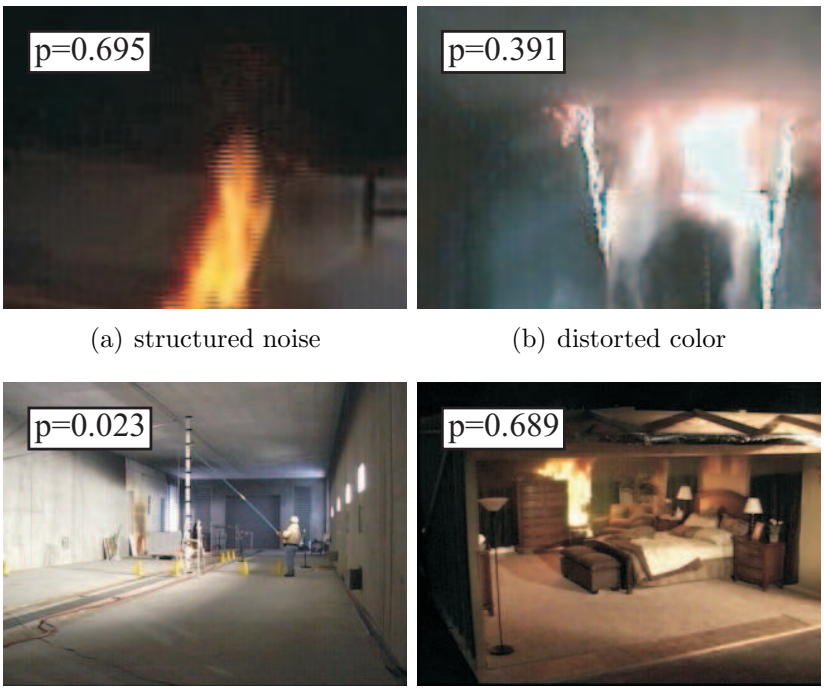
distorted to a blue tinge which causes the color transformation in Eq. (30) to assign low mass and thus, low motion magnitude. Third, insufficient spatial resolution, as in Figure 17(c), leads to very few pixels belonging to fire and motion structure cannot be detected. Lastly, partial occlusions may make the detector fail, as shown in Figure 17(d), because the occluding object's edge causes motion vectors along that edge to be strongly aligned in the direction of the image gradient. Line detection algorithms could be employed as a preprocessing step to exclude edge pixels from consideration.

Regarding false positive detections, two deficiencies are observed. In Figure 18(a), random noise on the red exit sign causes the motion field to have significant magnitude and turbulent directionality. Noise-reducing preprocessing can alleviate this problem, but other problems such as loss of texture detail may arise as a result. If spatial resolution is sufficient, multi-resolution techniques could be considered as well. Another false detection was observed for fast rotational motion (in combination with low spatial resolution) as seen in the case of orange print on the truck in Figure 18(b); the truck makes a sharp turn and undergoes a large change of perspective. This type of motion does not satisfy the intensity constancy assumption Eq. (22) as self-occlusions occur and intensity disappears during the rotation. In order to successfully reject such scenarios, a track-and-detect algorithm as proposed in [101] can monitor the scene

and set off alarm only if the anomaly persists.

2.6 Conclusion

The characteristic dynamics of flames have motivated the use of motion estimators to distinguish fire from other types of motion. Two novel optical flow estimators, OMT and NSD, have been presented that overcome insufficiencies of classical optical flow



(a) structured noise (b) distorted color

(c) resolution insufficiency (d) occlusion

Figure 17: Examples of false negative detections (Fire scene falsely classified as non-fire scene) with the resulting probabilities.

(a) noise (exit sign) (b) fast rotational motion

Figure 18: Examples of false positive detections (Non-fire scene falsely classified as fire scene) with the resulting probabilities.

models when applied to fire content. The obtained motion fields provide useful space on which to define motion features. These features reliably detect fire and reject non-fire motion, as demonstrated on a large dataset of real videos. Few false detections are observed in the presence of significant noise, partial occlusions, and rapid angle change.

The feature extraction scheme has been embedded in a general detection framework using active contours for tracking candidate regions, which allows the detection algorithm to operate more efficiently and robustly.

With regard to optical flow computations, future work may include the development of optical flow estimators with improved robustness to noise by taking into account more than two frames at a time. Further improvements of detection rates are expected when taking a multi-scale approach, since the resolution of the flames vary substantially between different scenarios. In order to build a full-scale flame detection software, the fusion of optical flow features with additional features such as color or spatio-temporal statistics is the most promising extension to explore in order to validate if these features complement each other.

CHAPTER III

COUPLED PHOTO-GEOMETRIC OBJECT FEATURES

3.1 Introduction

This chapter addresses vision-based object detection, localization, tracking, and recognition in infrared (IR) and thermal videos, which is a particularly challenging modality for standard computer vision algorithms as the image content exhibits less local structure compared to visible-spectrum imagery. The core idea of this chapter is to exploit Mumford-Shah-type energies [103], which are characterized by smooth radiance models for image regions (e.g., object and background). Since IR/thermal imagery is typically smoother than images in the visible spectrum, such models are expected to be particularly beneficial in the IR/thermal modality compared to low-level local features such as corners/edges, SIFT [92], or SURF [4]. From a mathematical modeling point of view, IR imagery and Mumford-Shah-type radiance approximations are both described by heat flow equations, which physically justifies the use of these methods.

Whereas one crucial aspect of flame detection in Chapter 2 is that flames—due to their unstructured and amorphous nature—demand local image analysis such as optical flow, the approach in this chapter is quite the opposite: Considering structured objects, such as pedestrians or vehicles, the goal of this chapter is to extract object shape and radiance information in an integrated and coupled manner. For such objects, shape is an intrinsic feature and is not affected by changes of illumination and radiance, which makes it a robust feature. However, shape can be ambiguous, for example in the presence of occlusions, and extracting the correct shape may not always be trivial. In such cases, it becomes essential to consider photometric features

as well. It is helpful to realize that geometry and radiance are intimately coupled through the physical object that creates the image: The physical object's geometry together with the optical (thermal) reflection (radiation) properties of its materials determine the object's boundary and appearance in the image. Changing the object's geometry (pose, location, deformation) results in predictable changes of the image radiance and the observed boundary shape.

The final goal of this chapter is the development of a recognition active contour that incorporates prior shape and radiance information about a target. To reach this goal, we start with a new segmentation energy (anisotropic Mumford-Shah) without priors, which yields an efficient radiance model in a limiting case. This radiance model will then be used for training along with global shape models in order to obtain coupled shape-plus-radiance priors.

3.1.1 Review

The developments in this chapter are derived from the classical Mumford-Shah energy [103], which is a region-based active contour that models radiance as piece-wise smooth functions where intensity discontinuities occur at the interface between object and background. Whereas edge-based active contours such as the geodesic active contour [78] "detect" these discontinuities by virtue of local gradients, the region-based approach relies on minimizing a region-based energy based on a certain image model. In the case of Mumford-Shah, the image model is chosen to be piece-wise smooth, but many other models can be and have been imagined. An extreme case of Mumford-Shah, for example, is the Chan-Vese energy [23], which models image regions as constants (maximally smooth). This energy can also be interpreted as separating the intensity mean inside and outside the contour. Similar models separating other one-dimensional statistics, such as image variance [129], higher-order statistics or even general density functions [79, 98] have been developed. Mumford-Shah-type

energies, which model images as smooth functions, are chosen for this project because they are expected to be particularly useful for images that are already comparatively smooth in nature, such as IR images.

In our development of anisotropic Mumford-Shah for image segmentation, the key idea is the *anisotropic* smoothing of the radiance function according to the underlying level set. Anisotropic smoothing (or more commonly known as anisotropic diffusion [108]) is extremely popular for image enhancement since the algorithm selectively smooths an image along edges, but not across edges, to preserve edges and remove pixel noise. In [108], the anisotropy of the gradient is determined by the image content, in particular edges. The motivation and approach for anisotropy in our research, however, is very different from [108]. Anisotropy will be defined in terms of the underlying level set representation of an object's shape, which will couple the anisotropy with the geometry of the object. This approach is motivated for two reasons. First, this type of smoothing is expected to be advantageous for image content with strong radial or angular patterns. Especially in medical images, we expect to deal with problems where the object has an onion-like structure, i.e., consists of different layers, so that the radiance information is primarily contained along the radial axis, not along the angular axis. The second motivation for anisotropic smoothing will be a limiting case where we average intensity along level sets, i.e., infinite smoothing along the level set's tangential direction. This approach leads to a one-dimensional (radial) representation of the radiance, which serves as an efficient, yet powerful radiance feature for principal component analysis (PCA) training. This one-dimensional radiance feature couples shape (level set) and radiance (intensity mean) information, which works towards our goal of coupling shape and radiance.

For the remainder of this review, we are concerned with the use of prior target information for segmentation. The use of prior information can considerably improve robustness in the presence of noise, occlusion, and model errors. A review for

recognition-segmentation approaches with different categories of priors, such as color, motion, shape, and texture is given in [32]. We will focus our attention on shape (geometric features) and radiance (photometric features) priors.

The authors of [65] provide a survey on shape-based models, in particular statistical models in medical imaging. A marker point representation for shape is introduced in [31] where a shape prior is obtained from a set of marker points that is manually labeled on the training data. By performing PCA on the locations of the marker points in different aligned training images, a finite set of modes is learned for the displacement of the marker points and the segmentation problem is posed in terms of the parameters of the training model. Another approach is given in [117], where the Fourier coefficients of the object boundaries represent shape and training is performed in this Fourier domain. A method where alignment of shapes is handled implicitly through the representation of shapes by Legendre coefficients of characteristic functions is proposed by [50]. For our research, the level set representation of shape is most relevant. Shape priors using level sets were pioneered by [88], who perform the PCA training on shape outlined in Section 3.3.2, which is then added as an additional term in the geodesic active contour energy to penalize dissimilarity of the active contour with the level set prior model. Shape, location and pose parameters of the prior are obtained by maximum a posteriori (MAP) estimation. In [24], a similar approach is taken, but parameter estimation is incorporated in the variational framework, whereas [88] pursue a statistical approach. Another approach for geodesic active contours is presented in [110]. In [18], geodesic and region-based contours are combined in the energy similar to [88], but in a purely variational formulation similar to [24]. The authors of [121] consider region-based energies such as [23] and [129] and minimize these energies with respect to PCA shape parameters as well as pose, so that the active contour is *restricted* to the training model, whereas in [88], the active contour is only penalized for deviations from the training model.

Further robustness can be achieved when combining shape and appearance information. Most famously, [30] extends the marker point approach in [31] to work with deformable appearance templates. A method similar to our research is shown in [128], where shape is represented as level sets, and PCA training is performed on the level sets as well as on the pixel-intensity of the training samples. Our approach is similar in that level set shape and intensity information is used for PCA training. In this chapter, however, we propose a smaller, one-dimensional set of intensity (radiance) features that, by definition, incorporate shape information. Our radiance model is, therefore, more efficient and already coupled to the geometry of an object. The works of [69, 51, 70, 115] extend [128], but all these approaches consider two-dimensional radiance models, which necessitate the use of warp functions and can be computationally complex and costly for this reason.

3.1.2 Overview

In the following sections, we first consider two segmentation energies in Section 3.2 that arise from the standard Mumford-Shah (MS) [103] when the radiance component is diffused anisotropically according to the underlying signed-distance level set function. We will call this new energy the *anisotropic* Mumford-Shah energy (aMS). The standard MS energy is first reviewed in Section 3.2.1, followed by the definition of the most general form of aMS in Section 3.2.2. A limiting case of general aMS, the iso-contour limiting case, is derived in Section 3.2.3.

The limiting case of aMS provides the mathematical motivation for the photo-geometric representation of an object in Section 3.3.1, which will be our main innovation towards coupled shape-plus-radiance priors for recognition-segmentation. The relevant training models for shape and radiance priors are presented in Sections 3.3.2 to 3.3.4, and finally, a recognition-segmentation active contour energy based on PCA models of level set shape and of the photo-geometric representation is proposed

and minimized in Section 3.4. Segmentation and tracking results are presented in Section 3.4.3 and conclusions are drawn in Section 3.5.

3.2 Anisotropic Mumford-Shah Segmentation Energies

3.2.1 Review: Mumford-Shah Energy

This section is a review of the standard MS energy [103], which will be important for understanding the subsequent developments. Given the segmentation for an image $I : \Omega \mapsto \mathbb{R}$ into foreground (object) region R and background region $R^c = \Omega \setminus R$, plus a radiance approximation $f : \Omega \mapsto \mathbb{R}$ for the image, the MS energy is given by

$$E_{MS}(C, f) = \alpha \int_{\Omega} (I - f)^2 \, d\Omega + \beta \left\{ \int_R \|\nabla f\|^2 \, d\Omega + \int_{R^c} \|\nabla f\|^2 \, d\Omega \right\} + \gamma \int_C 1 \, ds, \tag{43}$$

where in the first term (fidelity), the image I is compared in the least squares sense to the radiance model f. The second term (radiance smoothness) of the MS energy requires the radiance function to be smooth within each domain, thus leading to a piece-wise smooth radiance model, since there may be a radiance discontinuity at the interface $C = \partial R$ between R and R^c. Finally, the third term (curve smoothness) penalizes the length of the interface curve C for regularization of the contour. The constants α, β and γ are design parameters favoring data fidelity, radiance smoothness, or curve smoothness, respectively.

The optimal smooth radiance functions are obtained as solutions of the boundary value problems

$$\beta \Delta f^* = -\alpha (I - f^*) \quad \text{in } R, \qquad \text{with} \quad \frac{\partial f^*|_R}{\partial \mathcal{N}} = 0 \quad \text{on } C, \tag{44a}$$

$$\beta \Delta f^* = -\alpha (I - f^*) \quad \text{in } R^c, \qquad \text{with} \quad \frac{\partial f^*|_{R^c}}{\partial \mathcal{N}} = 0 \quad \text{on } C, \tag{44b}$$

which are damped versions of the Poisson equation (inhomogeneous Laplace equation). The Poisson equation is the steady state solution of the heat equation. In fact,

one may numerically solve Eq. (44) by discretely evolving the heat equation

$$f_t^* - \beta \Delta f^* = \alpha(I - f^*) \tag{45}$$

in R and R^c, respectively, with an arbitrary initial function (e.g., $f^*(t=0) = I$ or $f^*(t=0) = \text{const}$), while enforcing the Neumann boundary conditions along C.

The active contour for the MS is updated according to

$$C_t = \left(\alpha(I - f^*|_{R^c})^2 - \alpha(I - f^*|_R)^2 + \beta\|\nabla f^*|_{R^c}\|^2 - \beta\|\nabla f^*|_R\|^2 - \gamma\kappa\right)\mathcal{N}, \tag{46}$$

where κ and $\mathcal{N}$ are the curvature and the normal vector of the curve C, respectively. $f^*|_R$ ($f^*|_{R^c}$) denotes the optimal f^* restricted to R (R^c), and the derivatives are, therefore, to be taken as one-sided derivatives approaching the interface C from within the respective domain.

3.2.2 General Case

The classical MS functional could be called *isotropic* because the radiance function f is penalized by the norm of its gradient, which leads to isotropic smoothing of the original image. The gradient ∇ is independent of the choice of the coordinate system but is usually represented in the Cartesian image coordinate system. In general, however, the gradient can be locally represented by any basis. If, for example, we represent the curve C by the zero level set of a level set function ψ, we can decompose the gradient at each point into a component $\nabla_\perp$ perpendicular to the level set curve of ψ and a component $\nabla_\parallel$ parallel to the level set curve of ψ at that point, i.e., $\nabla = \nabla_\perp + \nabla_\parallel$ for each point in the image. With this choice of decomposing the gradient, the *anisotropic* Mumford-Shah (aMS) energy is proposed next by separate penalization of the gradient components $\nabla_\perp$ and $\nabla_\parallel$,

$$\begin{aligned} E_{\text{aMS}}(C, f) = \alpha \int_\Omega (I - f)^2 \, d\Omega + \beta_\perp^{in} \int_R \|\nabla_\perp f\|^2 \, d\Omega + \beta_\parallel^{in} \int_R \|\nabla_\parallel f\|^2 \, d\Omega + ... \\ + \beta_\perp^{out} \int_{R^c} \|\nabla_\perp f\|^2 \, d\Omega + \beta_\parallel^{out} \int_{R^c} \|\nabla_\parallel f\|^2 \, d\Omega + \gamma \int_C 1 \, ds. \end{aligned} \tag{47}$$

The parameters $\beta_{\perp}^{in}$, $\beta_{\perp}^{out}$ and $\beta_{\parallel}^{in}$, $\beta_{\parallel}^{out}$ are the weights for the two gradient components on R and R^c, respectively. Choosing $\beta_{\perp}^{in} = \beta_{\perp}^{out} = \beta_{\parallel}^{in} = \beta_{\parallel}^{out} = \beta$ returns the standard (isotropic) MS functional by virtue of the Pythagorean theorem. The aMS energy is, therefore, a generalization of the standard isotropic MS energy, and may, therefore, be suitable for problems that the standard MS functional is not.

Before providing an example and further motivation, the minimization of the aMS energy is presented for better understanding of the proposed energy. The full derivation can be found in Appendix A.1. For ease of presentation, we only show the terms contributing from the object region R, noting that the terms from the background region R^c are obtained analogously. In other words, in the following paragraphs, f denotes $f|_R$, $\beta_{\perp}$ is $\beta_{\perp}^{in}$, $\beta_{\parallel}$ is $\beta_{\parallel}^{in}$, and the analogous terms derived from the integral terms over R^c (plus the standard curve smoothness term) would have to be added to obtain the full gradients. Joint minimization using calculus of variations leads to the boundary value problem

$$-\alpha(I - f^*) - \beta_{\perp}\nabla \cdot \nabla_{\perp} f^* - \beta_{\parallel}\nabla \cdot \nabla_{\parallel} f^* = 0 \quad \text{in } R, \tag{48a}$$

$$(\beta_{\perp}\nabla_{\perp} f^* + \beta_{\parallel}\nabla_{\parallel} f^*) \cdot \mathcal{N} = 0 \quad \text{on } C, \tag{48b}$$

for the optimal radiance function f^* given C. The resulting radiance model from Eq. (48) is illustrated in Figure 19 for a synthetic image, where radiance functions for the image in the top left corner are shown for various pairs of $\beta_{\perp}$ and $\beta_{\parallel}$. The diagonal line is equivalent to isotropic MS with the limiting case of Chan-Vese [23] (Mumford-Shah with constant radiance approximation) for $\beta_{\perp} = \beta_{\parallel} \to \infty$. The off-diagonal images correspond to smoothing predominately in the angular (top right) and radial (bottom left) directions. The aMS has the ability to model radial or angular patterns in an image, whereas MS ignores these patterns by smoothing in all directions.

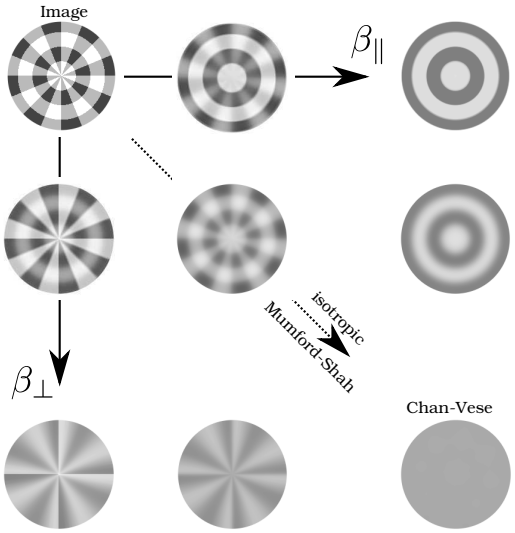


Figure 19: Example for the radiance model for anisotropic Mumford-Shah varying $\beta_{\perp}$ and $\beta_{\|}$.

Regarding the minimization with respect to the curve, since the radiance smoothness penalty is now coupled to the level sets ψ of the curve C (as ψ determines the directionality of the gradient components $\nabla_{\perp}$ and $\nabla_{\|}$ at each image point), deriving the shape gradient becomes considerably more complicated compared to the isotropic MS. In order to keep the problem tractable, it is assumed that ψ is an L_2-signed-distance function, which allows to analytically connect the evolution of the curve C to changes in the level set function ψ. Maintaining ψ to be an L_2-signed-distance function can be efficiently achieved by fast marching methods as described in [114]. The partial curve flow (stemming from the integrals over R) is then found to be

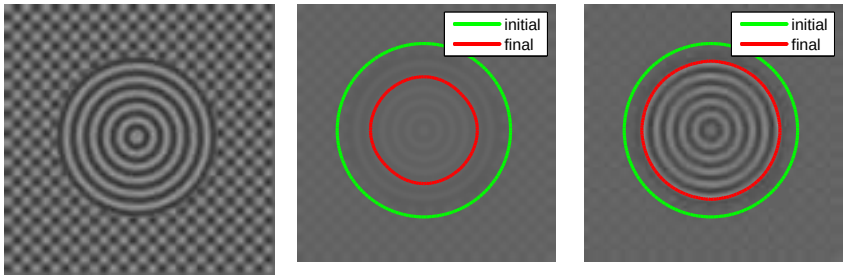
$$\begin{aligned} C_t^{in} = [&-\alpha(I-f^*)^2 - \beta_{\perp}\|\nabla_{\perp}f^*\|^2 - \beta_{\|}\|\nabla_{\|}f^*\|^2 + \dots \\ &+ 2(\beta_{\perp}-\beta_{\|}) \int_{\tau_{min}(s)}^{0} \frac{\partial}{\partial s_\tau} \left(\frac{\partial f^*}{\partial \tau} \frac{\partial f^*}{\partial s_\tau} \right) (1+\kappa^*\tau)\, d\tau]\, \mathcal{N}, \end{aligned} \tag{49}$$

where s_τ and τ are local coordinates parametrizing the domain R in an angular and a radial component, respectively, and κ^* denotes the curvature along C. $\tau_{min}(s)$ is the negative distance from a curve point to the skeleton of the level set function. The last integration term arises from the fact that f^* depends on the level set function ψ,

which in turn is coupled to the curve C. This last term makes the implementation of general aMS cumbersome, as a line integral has to be computed for each point on the curve C. Moreover, stability of this flow may be an issue as the sign of the integrand may change depending on the data leading to cases resembling backward heat-flow, which is well known to be numerically unstable.

General anisotropic Mumford-Shah is not expected to always outperform standard Mumford-Shah, but is rather expected to perform favorably for certain types of images, in particular when the foreground or background is radially or angularly structured. A synthetic example where aMS is superior to MS is shown in Figure 20(a). The object has a clear radial pattern, which is smoothed out by the MS in Figure 20(b) making the distinction between object and background difficult. For the aMS result in Figure 20(c), the parameters are chosen as $\beta := \beta_{\perp}^{out} = \beta_{\parallel}^{in} = \beta^{out}$, and $\beta_{\perp}^{in} < \beta$, thus preserving the radial structure of the inner circular object, which eventually leads to successful segmentation. This preliminary result suggests that the aMS might be a valuable extension to existing MS methods where the radiance pattern of the object follows a clear radial or angular pattern, e.g., in medical images where different layers of tissues create images with strong gradients across layers but low gradients within layers.

While we leave further exploration and analysis of the general aMS energy to future research, the main focus of this chapter is a special case of the general aMS



(a) original image (b) modeled image MS (c) modeled image aMS

Figure 20: Anisotropic Mumford-Shah versus isotropic Mumford-Shah.

energy, which eventually leads to a one-dimensional object radiance model that is used to obtain efficient training models for classes of objects. Towards this goal, a special, limiting case of the general aMS energy is presented in the following section.

3.2.3 Iso-Contour Limiting Case

In this section, the case $\beta_{\|} \to \infty$ is presented. Again, the convention is to only consider the terms coming from integrals over R in the aMS functional Eq. (47) (noting that the background terms can be obtained analogously) and the same notation applies. Letting $\beta_{\|} \to \infty$ implies that f must be constant along each connected component of each level set of ψ. If each level set consists of a single component, this means f will have the same level sets as ψ, though not necessarily with matching values. If we impose this as a constraint on f, regardless of the connectivity of the level sets of ψ, then the function $f : R \mapsto \mathbb{R}$ may be written as

$$f(x) = \mathrm{f}(\psi(x)), \tag{50}$$

where $\mathrm{f} : \mathbb{R} \mapsto \mathbb{R}$. This radiance model will be called "iso-contour radiance model" because it is constant along the object's contours. For notational convenience, we rename $\beta := \beta_{\perp}$, and restate the interior part of the aMS energy as

$$\begin{aligned} E^{in}_{\mathrm{aMS}}(C, \mathrm{f}) &= \int_R \alpha(I - \mathrm{f} \circ \psi)^2 + \beta \|\nabla_{\perp}(\mathrm{f} \circ \psi)\|^2 dx \\ &= \int_R \alpha(I - \mathrm{f} \circ \psi)^2 + \beta \left(\dot{\mathrm{f}} \circ \psi\right)^2 \|\nabla \psi\|^2 dx. \end{aligned} \tag{51}$$

The full derivation of the minimizing equations is presented in Appendix A.2. Here, the results are summarized next. As in the general case of aMS, it is assumed that ψ is a L^2-signed-distance function, so it uniquely defines a family of inner curves

$$C_T = \{x \in R : \psi = T\}, \tag{52}$$

the T-level-sets of ψ, where T takes values from ψ_{min} (innermost object point) to 0 (boundary curve). Thus, we may use T as a "radial" variable (analogous to the

variable τ in Appendix A.1.3), noting that T is actually the value of the level set function ψ. For a given ψ, the one-dimensional radiance function $\mathrm{f}(T)$ is, therefore, defined on the interval $[\psi_{min}, 0]$, and the optimal radiance function $\mathrm{f}^*(T)$ for the aMS functional in the iso-contour limiting case satisfies the Euler-Lagrange equations

$$\alpha(m_1\mathrm{f}^* - m_2) - \beta(m_3\ddot{\mathrm{f}}^* + \dot{m}_3\dot{\mathrm{f}}^*) = 0 \quad \text{for } T \in [\psi_{min}, 0], \tag{53a}$$

$$\dot{\mathrm{f}}^*(\psi_{min}) = \dot{\mathrm{f}}^*(0) = 0, \tag{53b}$$

where $\dot{(\cdot)}$ denotes the derivative with respect to T and

$$\begin{aligned} m_1(T) &= \int_{C_T} \frac{1}{\|\nabla\psi\|}\, ds, \\ m_2(T) &= \int_{C_T} \frac{I(x)}{\|\nabla\psi\|}\, ds, \\ m_3(T) &= \int_{C_T} \|\nabla\psi\|\, ds. \end{aligned} \tag{54}$$

For generality, we have not yet made use of the fact that $\|\nabla\psi\| = 1$ everywhere for a signed-distance function. The quantities m_1, m_2, m_3, and $\dot{m}_3$ can be efficiently computed using the marching method derived in Section 3.2.4. The resulting ODE with boundary conditions Eq. (53) is solved by discretely evolving the parabolic flow equation

$$\mathrm{f}^*_t = \alpha(m_2 - m_1\mathrm{f}^*) + \beta(m_3\ddot{\mathrm{f}}^* + \dot{m}_3\dot{\mathrm{f}}^*) \tag{55}$$

to steady state, using finite differences to approximate derivatives in T and artificial time t. The optimal curve flow is found in Appendix A.2 to be

$$C_t^{in} = \left[-\alpha(I - \mathrm{f}^*(0))^2 + 2\int_{\tau_{min}(s)}^{0} \frac{\partial \mathrm{f}^*}{\partial T}\left(-\alpha(I - \mathrm{f}^*) + \beta\frac{\partial^2 \mathrm{f}^*}{\partial T^2}\right)(1 + \kappa^*\tau)\, d\tau\right]\mathcal{N}, \tag{56}$$

where κ^* is the curvature of C. Again, the last integration term arises from the fact that f^* depends on the level set function ψ, which in turn is coupled to the curve C. Compared to the general case of aMS, this limiting case is numerically more efficient and stable, since the problem essentially becomes one-dimensional in the radiance component. The remaining parameter $\beta := \beta_\perp$ controls the smoothing across level

sets. The larger β, the more image information is smoothed to the boundary and the limiting case of Chan-Vese is obtained for $\beta \to \infty$.

In the extreme case that $\beta = 0$ and using the assumption that ψ is a signed-distance function, the optimal radiance is easily obtained from Eq. (53) as

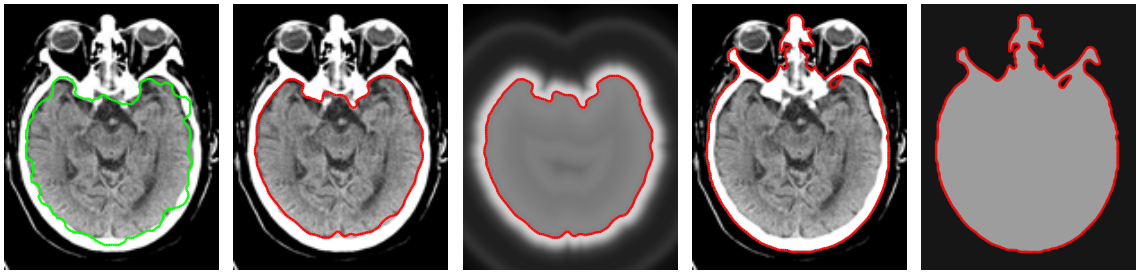
$$\mathrm{f}^*(T) = \frac{\int_{C_T} I(x)\, ds}{\int_{C_T} 1\, ds} = \frac{\int_{C_T} I(x)\, ds}{\mathrm{length}(C_T)}, \tag{57}$$

meaning that $\mathrm{f}^*(T)$ is the image intensity mean along the level set curve C_T. Note, therefore, that according to Eq. (56), choosing $\beta = 0$ results in a zero fidelity term in the curve gradient, since then $\mathrm{f}_{in}(0) = \mathrm{f}_{out}(0)$, which cancels the fidelity term (due to object and background terms appearing with opposite signs in the total gradient). Therefore, $\beta > 0$ is required for a meaningful segmentation energy. The case $\beta = 0$ will be, however, of interest in Section 3.3.1, where $\mathrm{f}^*(T)$ will define a photo-geometric representation of an object that can be trained on using principal component analysis.

A promising segmentation result using the iso-contour radiance model for $\beta > 0$ is shown for the CT image of a skull in Figure 21. Assuming we want to segment the brain from the skull in Figure 21(a), the aMS with iso-contour radiance provides the desired segmentation in Figure 21(b) whereas the Chan-Vese [23] active contour, which models object and background as constant, spills in the skull region in Figure 21(d). This segmentation difference is explained by the radiance models for aMS in Figure 21(c) and CV in Figure 21(e): The aMS has local behavior because it is less diffusive than the Chan-Vese energy, which diffuses to steady state (constant). Both energies have their advantages in certain scenarios, but for layer-shaped objects as they often appear in medical images, the aMS is expected to be superior due to its explicit modeling of radial radiance functions.

3.2.4 Intrinsic Integration

For the efficient computation of the numerical integrals in Eq. (54) for a discrete set of T-level-sets, a marching method is devised, which makes use of the fact that the level



(a) initial contour (b) aMS (c) aMS model (d) CV (e) CV model

Figure 21: Anisotropic Mumford-Shah (aMS) with iso-contour radiance versus Chan-Vese (CV) (Image by courtesy of `http://www.med.wayne.edu/`).

set function is a signed-distance function, which (for the object region) decreases smoothly as we move away from the zero level set. We are faced with computing integrals of the form

$$m(T) = \int_{C_T} h \, ds \tag{58}$$

for a generic function $h : \mathbb{R}^2 \mapsto \mathbb{R}$. The method of explicit level set extraction is not practical because our goal is to compute an entire family $m(T)$ of integrals, which makes the explicit extraction of level sets very inefficient. An alternative is to first construct the anti-derivative $M(T)$

$$M(T) = \int_{-\infty}^{T} \int_{C_\tau} h \, ds \, d\tau = \int_{\psi(x)<T} h \|\nabla \psi\| \, dx, \tag{59}$$

which, by the coarea formula, is obtained by volumetric integration of $h\|\nabla\psi\|$ over the progressively expanding volume $\{x : \psi(x) < T\}$. Then, $m(T) = \dot{M}(T)$ is obtained by differentiation.

Approximating h to be constant and ψ to be linear (so that the gradient is constant) in a pixel cell, we write

$$M(T) \approx \sum_i M_i(T) = \sum_i h(x_i) \, \|\nabla \psi(x_i)\| \, V_i(T), \tag{60}$$

where x_i are the grid (cell) center points and $V_i(T)$ are partial volumes defined as

$$V_i(T) = \int_{\text{cell}(\Delta x, x_i)} H\left(T - \psi_i(x)\right) dx, \tag{61}$$

where H denotes the Heaviside function, Δx is a vector of cell widths in x and y, and $\psi_i(x)$ is the linear approximation

$$\psi_i(x) = \psi(x_i) + (x - x_i) \cdot \nabla\psi(x_i), \quad x \in \text{cell}(\Delta x, x_i). \tag{62}$$

These partial volumes can be shown to simplify to

$$V_i(T) = \left(\prod \Delta x\right) v_i(\tau, d), \tag{63}$$

where $\prod \Delta x$ is the product of the cell widths, and

$$v_i(\tau, d) = \begin{cases} 0, & \tau \leq 0 \\ \frac{1}{2}\frac{\tau^2}{d(1-d)}, & 0 < \tau < d \\ \frac{1}{2} + \frac{\tau - \frac{1}{2}}{1-d}, & d < \tau < 1-d \\ 1 - \frac{1}{2}\frac{(1-\tau)^2}{d(1-d)}, & 1-d < \tau < 1 \\ 1, & \tau \geq 1 \end{cases} \tag{64}$$

where τ and $0 < d \leq 0.5$ are pre-computed quantities of the level set function ψ. Since the volume $\{x : \psi(x) < T\}$ is increasing with T, so is $V_i(T)$. Thus, once $V_i(T)$ has become $\prod \Delta x$ for some T, it will remain $\prod \Delta x$ for the rest of the algorithm, and we can store its contribution without recomputing it at each step. Also, when assuming that ψ is a signed-distance function, we know that only a close neighborhood around a pixel with $0 < V_i(T) < \prod \Delta x$ will become non-zero for the next higher T, which makes the algorithm even faster.

3.3 Shape-Plus-Radiance PCA Training Models

This section works towards the formulation of the contour energy in Section 3.4 that includes prior object shape and radiance knowledge to make segmentation more robust and efficient. The use of target training data is also beneficial in tracking, detection, and recognition tasks, even though the latter two will not be addressed

in thesis. The object training models presented next are based on principal component analysis (PCA) of shape and radiance. In short, PCA is performed on shape (level set function) [121] in Section 3.3.2 and a one-dimensional radiance function (called *photo-geometric representation*) extracted from labeled object training data in Section 3.3.3. The photo-geometric representation is introduced in Section 3.3.1 as the radiance model of the aMS iso-contour limiting case where $\beta = 0$. In Section 3.3.4 the coupled shape and radiance PCA is described, exploits correlations between shape and radiance in the training set, and reduces, therefore, the number of required principal components and unknown parameters by half.

The resulting PCA models are truncated to a certain number of principal components with the largest eigenvalues form a lower dimensional linear model in this features space. This truncation ensures that the dominant characteristics among training samples is preserved whereas small deviations and disturbances pertaining to isolated training samples are discarded, making the models robust and efficient. These PCA training models are used in Section 3.4, where a segmentation energy similar to the aMS limiting case is minimized with respect to the principal component weights, thus effectively restricting the contour and the modeled radiance to a linear subset of the training data.

3.3.1 Shape and Photo-Geometric Representation of an Object

Given the boundary shape C (and the corresponding signed-distance level set function ψ representing C) as well as the radiance intensities I of an object, the photo-geometric representation of an object is defined to be the aMS radiance model in the iso-contour case with $\beta =$, i.e.,

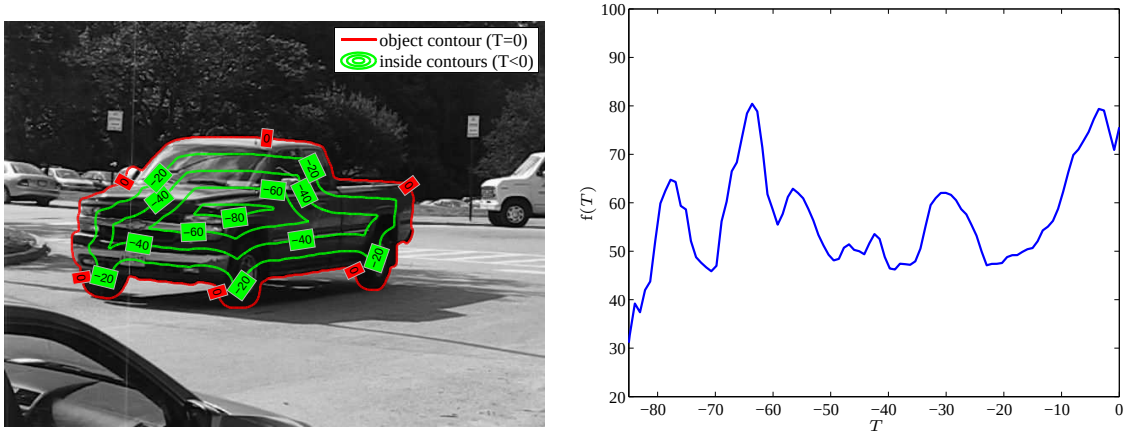
$$\mathrm{f}(T) = \frac{\int_{C_T} I \, ds}{\int_{C_T} 1 \, ds} = \frac{\int_{C_T} I \, ds}{\mathrm{length}(C_T)}, \tag{65}$$

where $\mathrm{f} : [\psi_{\min}, 0] \mapsto \mathbb{R}$ are the mean intensities along the level sets $C_T = \{x \in R : \psi(x) = T\}$. In Figure 22, these quantities are illustrated for the image of a truck from

the Berkeley Motion Segmentation dataset [19], where ground truth segmentation of the truck is provided. We first compute the L^2-signed-distance function ψ for the object shape. The level sets for some values of T between $\psi_{\min}$ and 0 are shown in Figure 22(a). The mean intensity of the radiance along these curves is then computed using intrinsic integration from Section 3.2.4 and plotted in Figure 22(b).

The term "photo-geometric" is chosen because f(T) couples the object's geometric information C_T with its photometric information I, and thus attempts to capture the physical coupling between geometry and radiance of rigid objects. As a feature, it is well-suited for training purposes because it is invariant to translation, and rotation of the object. It is also invariant to scale if the domain of f is scaled to a constant interval, e.g., $[-1, 0]$.

As an appearance model, the photo-geometric representation can be viewed as a compromise between the competing goals of robustness, efficiency, and discrimination power. Active appearance models [30], which are templates that are warped and matched with the help of marker points, present a powerful, yet cumbersome method for radiance modeling. For controlled environments, these methods work effectively, but the amount of required training data quickly becomes tremendous.



(a) level set curves C_T (b) mean intensities along T-level-sets f(T)

Figure 22: Illustration of the photo-geometric representation.

Two-dimensional generative radiance models, such as for example in [71, 72], need to be warped by diffeomorphisms to handle changing object shapes, which is a cumbersome process. In [36, 37]. Another example with two-dimensional radiance model is [128], who train on coupled shape and radiance vectors. On the other end, there are simple methods, which model object radiance simply as a constant, e.g., Chan-Vese [23], or more generally through a finite set of statistics. While more robust and tractable, these methods may not be accurate enough to model more complicated object radiance functions and spatial distribution of intensity values. Our proposed integrated shape-plus-radiance algorithms using a one-dimensional photo-geometric descriptor is more general than finite sets of statistics, and at the same time not as cumbersome as two-dimensional radiance models that require warps to migrate between different configurations. In our approach, the integration of shape and radiance is naturally built in in the definition of the features, yet still an infinite dimensional radiance descriptor (a one-dimensional function) is kept, which has the capability of producing accurate results.

3.3.2 Shape PCA

The goal of this section is to design a shape training model from shapes of similar objects (the training data). By "shape", we refer to the boundary curve of an object as it is seen in an image. Since the boundary curve is equivalent to the level set function of that curve when the level set function is chosen to be an L^2-signed-distance function, this training model is derived in the level set framework based on principal component analysis (PCA) on the level set functions of the training objects. This work was performed in [121], and is summarized next. In subsequent section, we extend this shape-based model by radiance models for the training object.

We assume training shapes to be initially given binary label images. If the shape is given as a level set, the binary image I_b is simply obtained as $I_b = 1$ where $\psi \leq 0$,

and $I_b = 0$ where $\psi > 0$. Each binary image is aligned by a parametric transformation (translation, rotation, and scale) to create as much overlap as possible between the training labels. This overlap is measured by the energy functional

$$E_{\text{align}} = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{\int\int_{\Omega} (\tilde{I}_b^i - \tilde{I}_b^j)^2 dx}{\int\int_{\Omega} (\tilde{I}_b^i + \tilde{I}_b^j)^2 dx}, \tag{66}$$

where $\tilde{I}_b^i$, $i = 1, ..., N$ are the transformed training labels. The minimization procedure for the alignment energy is detailed in [121] and follows standard principles of variational methods (computation of the gradient, followed by the gradient descent method). After alignment, the transformed binary images are converted to L^2-signed-distance functions.

Assuming that shapes of a class can be represented approximately as a linear combination of the training shapes, the following PCA training model is justified to represent class shapes: First, the $m \times n$ level set function ψ_i of the i-th training shape is represented as a mn-dimensional vector $\boldsymbol{\psi}_i$ by stacking the columns on top of each other to form one long column vector. Then, the mean $\bar{\Phi} = (1/N) \sum \boldsymbol{\psi}_i$ is subtracted to obtain zero-mean training vectors $\tilde{\boldsymbol{\psi}}_i = \boldsymbol{\psi}_i - \bar{\Phi}$. Finally, the training matrix $\mathcal{S}$ is obtained by stacking the N training vectors next to each other, $\mathcal{S} = [\tilde{\boldsymbol{\psi}}_1, \tilde{\boldsymbol{\psi}}_2, \cdots, \tilde{\boldsymbol{\psi}}_N]$. The eigenvalue decomposition of the matrix $(1/N)\mathcal{S}^T\mathcal{S}$ provides N eigenvalues d_i and eigenshapes Φ_i, only a few of which represent most of the training data when the training shapes are similar. These $K < N$ eigenshapes are retained to build the training model

$$\Phi(x; \boldsymbol{w}) = \bar{\Phi}(x) + \sum_{i=1}^{K} w_i \Phi_i(x), \tag{67}$$

where $\boldsymbol{w}$ are weights to be determined.

For a given new shape, the best approximation through the PCA training model is given by first aligning the new shape to the training set as described above (while fixing the already aligned training samples) and then projecting the level set function

ψ of the aligned test label onto the training model Eq. (67) according to

$$w_i = \langle \psi, \Phi_i \rangle, \quad i = 1, \ldots, K, \tag{68}$$

where $\langle \cdot, \cdot \rangle$ is the l_2 inner product. The residual

$$r(\boldsymbol{w}) = \|\Phi(x; \boldsymbol{w}) - \psi(x)\| \tag{69}$$

quantifies the dissimilarity between the new shape and the training model, meaning that the smaller r, the better the new shape is represented by the training model Eq. (67).

3.3.3 Radiance PCA

Analogously to shape PCA, PCA may be performed on the photo-geometric representation f(T) Eq. (65) of an object. Owing to the photo-geometric representation being invariant to translation, scale, and rotation, alignment is not needed. The resulting radiance model—by radiance, we mean photo-geometric representation f(T)—is, analogous to shape PCA above,

$$\mathrm{F}(\tau; \boldsymbol{w}) = \bar{\mathrm{F}}(\tau) + \sum_{i=1}^{K} w_i \mathrm{F}_i(\tau), \tag{70}$$

where $\bar{\mathrm{F}}$ is the mean photo-geometric representation, and F_i are the "eigenradiances".

3.3.4 Coupled Shape-Plus-Radiance PCA

In order to exploit both shape and radiance for the training model, shape and radiance PCA can be performed independently leading to a $2k$-dimensional shape and radiance training model

$$\Phi(x; \boldsymbol{w}) = \bar{\Phi}(x) + \sum_{i=1}^{K} w_i \Phi_i(x), \tag{71a}$$

$$\mathrm{F}(\tau; \boldsymbol{v}) = \bar{\mathrm{F}}(\tau) + \sum_{i=1}^{K} v_i \mathrm{F}_i(\tau), \tag{71b}$$

where the weights $\boldsymbol{w}$ and $\boldsymbol{v}$ are independent since PCA is performed independently on shape and radiance.

Alternatively, by combining shape and radiance in one PCA procedure, an equally good or better training model is expected since correlations between shape and radiance can be exploited. Moreover, the resulting training model will have half as many parameters as the decoupled training model Eq. (71). The coupling is achieved by performing PCA on the Cartesian product of shape ψ and photo-geometric representation f(T) of objects, that is by stacking the shape vector and the radiance vector into one column vector Λ and performing PCA on this vector. The resulting training model is

$$\Lambda(x, \tau; \boldsymbol{w}) = \bar{\Lambda}(x, \tau) + \sum_{i=1}^{K} w_i \Lambda_i(x, \tau), \tag{72}$$

which, upon de-stacking $\Lambda(x, \tau; \boldsymbol{w}) = (\Phi^{\text{coupled}}(x; \boldsymbol{w}), \mathrm{F}^{\text{coupled}}(\tau; \boldsymbol{w}))$, yields

$$\Phi^{\text{coupled}}(x; \boldsymbol{w}) = \bar{\Phi}^{\text{coupled}}(x) + \sum_{i=1}^{K} w_i \Phi_i^{\text{coupled}}(x), \tag{73a}$$

$$\mathrm{F}^{\text{coupled}}(\tau; \boldsymbol{w}) = \bar{\mathrm{F}}^{\text{coupled}}(\tau) + \sum_{i=1}^{K} w_i \mathrm{F}_i^{\text{coupled}}(\tau), \tag{73b}$$

where the same weights $\boldsymbol{w}$ apply. Note that the eigenvectors and means are different from the decoupled case in Eq. (71) as the coupled PCA procedure computes the eigenvectors in the combined shape-radiance space.

3.3.5 Experiments

This section explores how a coupled shape-plus-radiance PCA model compares to the decoupled case in a synthetic reconstruction experiment. Under the assumption that shape and radiance are highly correlated, it is expected that a coupled shape-plus-radiance training model will achieve equal or better reconstruction performance from heavily corrupted test images, while it essentially cuts the dimensionality of the training model in half. For these experiments, it is assumed that the target object shape and location is provided as ground truth.

We consider the training images in Figure 23, which have similar "fighter" shapes and similar photo-geometric representations as shown in Figure 24 for a few training images. Training model "separate" (TM-separate) is obtained by separately performing PCA on shape and the photo-geometric representation f(T) as explained in the previous sections. Training model "coupled" (TM-coupled) is obtained by combining the level set functions (shape) and photo-geometric features (radiance) in one large feature vector, and performing PCA on this combined feature vector as shown in Section 3.3.4. For reconstruction, we keep the 6 largest out of the 12 principal components, that is TM-separate has 12 parameters (6 for shape and 6 for radiance), and TM-coupled has 6 parameters. In the following three experiments, we corrupt the quality of the test images by

1. additive noise on radiance, but no shape corruption,

2. occluded shape, but no radiance corruption,

3. both additive noise and occlusion,

and then evaluate the reconstruction performance of TM-separate and TM-coupled.

3.3.5.1 Test 1: Corrupted Radiance

In all experiments, test images are chosen by the leave-one-out cross-validation method, meaning that one of the ground-truth images is always left out from training and tested on. The original test image (the left-out training image) is corrupted by zero-mean uniformly distributed additive noise, and the image is then thresholded for pixel values to lie between $[0, 255]$, which adds non-linear behavior to the corruption mechanism. In Figure 25, some test images are displayed for different noise levels.

Figure 23: Training data for coupled shape-plus-radiance PCA experiment.

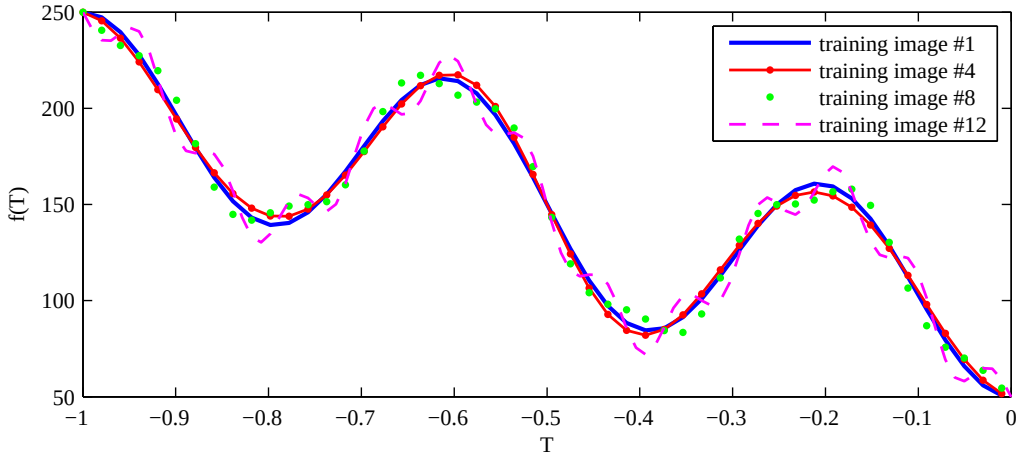


Figure 24: Photo-geometric representation of the training data.

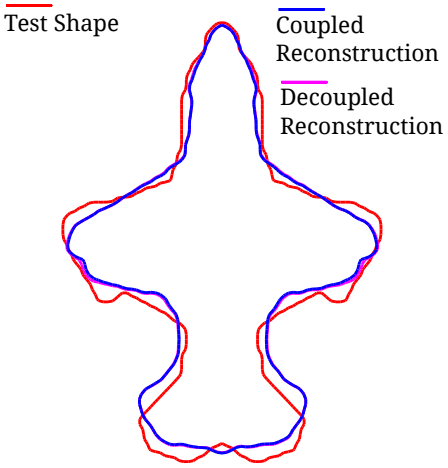
(a) noise=10 (b) noise=50 (c) noise=100 (d) noise=150 (e) noise=200 (f) noise=300

Figure 25: Example test image for test 1 with radiance corrupted by varying noise levels.

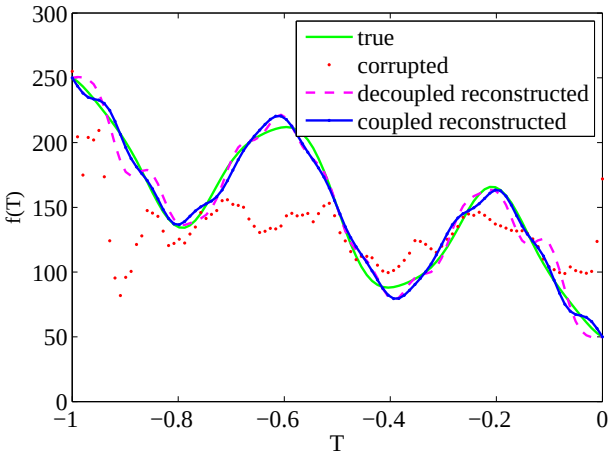
In the reconstruction phase, the corrupted test image is projected onto TM-separate and TM-coupled (for shape and f(T), respectively), and the projected (reconstructed) image is compared to the original (not corrupted) test image. Figure 26 shows an example of a reconstruction with very strong noise. Apart from the visually appealing reconstruction of both shape and radiance using TM-separate and TM-coupled, another noteworthy observation is that the photo-geometric representation is very robust to zero-mean additive pixel noise. For a human observer, the corrupted radiance does not appear similar at all to the true radiance pattern, but in the photo-geometric representation, the corrupted (red) f(T)-function looks recognizably similar to the true (green) function.

The results of these random (due to the randomness of the added noise) experiments are averaged over many repetitions and the resulting average PCA residuals

(a) original image (b) corrupted image (c) reconstructed image (decoupled) (d) reconstructed image (coupled)



(e) shape



(f) radiance

Figure 26: Example reconstruction with noise level 300.

(L_2-distance between the reconstructed and the original test image) for shape and radiance are shown in Figure 27. Inspecting the radiance plot, it is observed that TM-coupled outperforms TM-separate when noise levels become high, whereas for small noise levels, decoupled PCA performs better. The performance increase of TM-separate for high noise levels is paid only by a small penalty in the shape component, which makes TM-coupled more suitable for heavy-noise environments.

The reason why TM-coupled is more suitable for noisy environments becomes more obvious when we include the original test image in the training data and run the same experiment. Now, TM-coupled can directly correlate the correct test shape with the true radiance of the training sample, and reconstruction of the radiance becomes very accurate as is shown in Figure 28. Even when the test image is not included in the training, the training shapes and radiances of the training data are

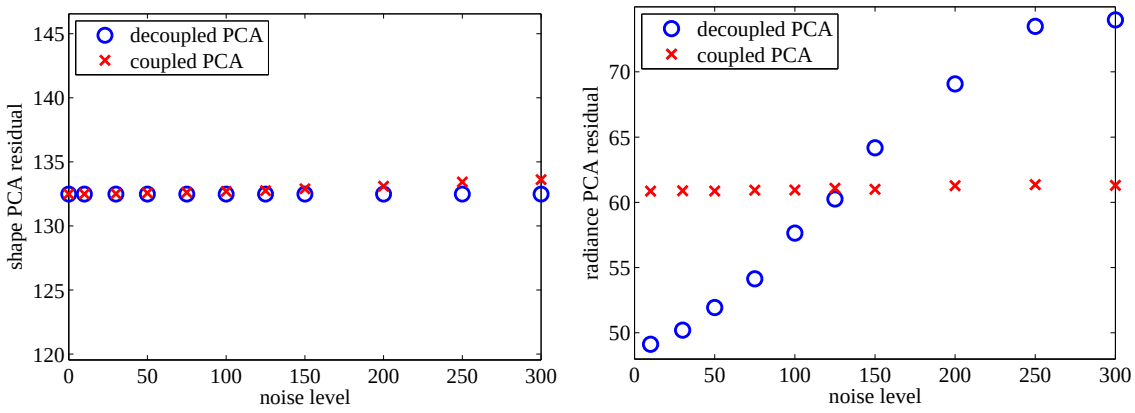


Figure 27: Test 1 result: PCA residuals over noise level.

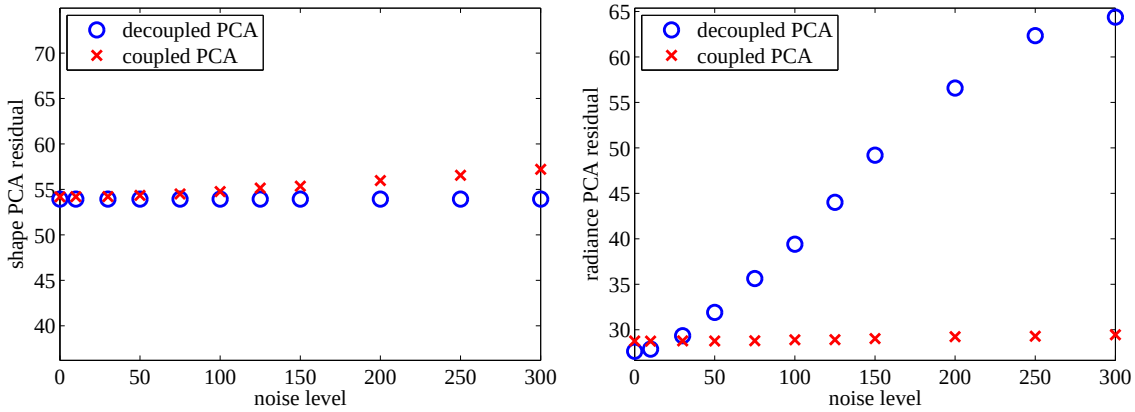


Figure 28: Test 1 result: PCA residuals over noise level when original test image in training set.

sufficiently correlated for TM-coupled to make use of this correlation despite the radiance being strongly corrupted.

3.3.5.2 Test 2: Corrupted Shape

In the second experiment, we leave the pixel radiance of the test object untampered, but increasingly occlude the object from the top as is shown in Figure 29. Note that shape corruption implies corrupted photo-geometric representations because the f(T) are computed along the level sets of the object's shape. We perform the same reconstruction experiment as in Test 1, but now varying the occlusion level instead of additive noise. The resulting PCA residuals are shown in Figure 30.

(a) 15% (b) 20% (c) 25% (d) 30% (e) 40% (f) 50%

Figure 29: Example test image for test 2 with varying occlusion levels (top x% of the image occluded).

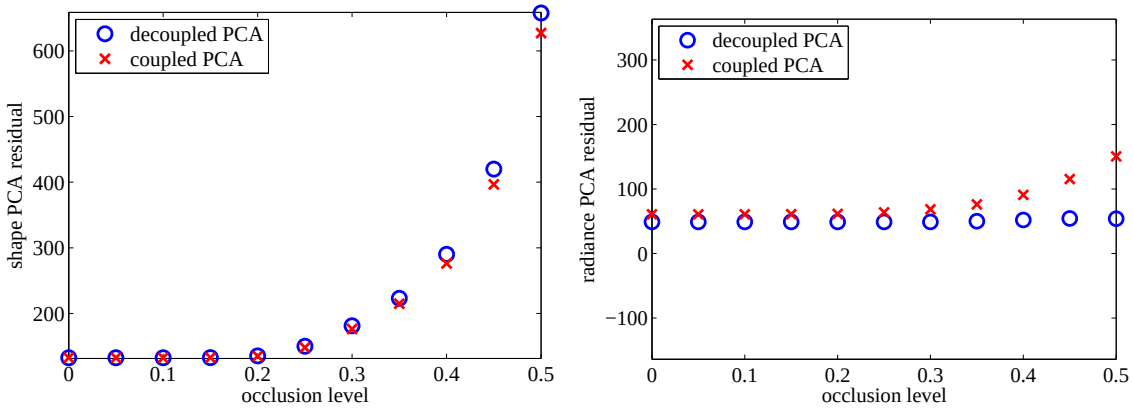


Figure 30: Test 2 result: PCA residuals over occlusion levels with no additive radiance noise.

It is seen that TM-coupled slightly outperforms TM-separate in shape reconstruction, but TM-separate beats TM-coupled in terms of radiance for strong occlusion. This experiment shows that relatively similar performance can be obtained with half as many training components by using the coupled shape-plus-radiance model, whereas Test 1 is a case with not just equivalent, but even better performance with half as many training parameters.

3.3.5.3 Test 3: Corrupted Shape and Radiance

In the final experiment, shape and pixel radiance corruption are combined. The test image is corrupted by a medium occlusion level of 25% in addition to varying noise levels as shown in Figure 31. Again, the same reconstruction method is performed as in the previous two tests, an example of which is presented in Figure 32. Both shape and radiance are well recovered using TM-separate and TM-coupled. In terms of

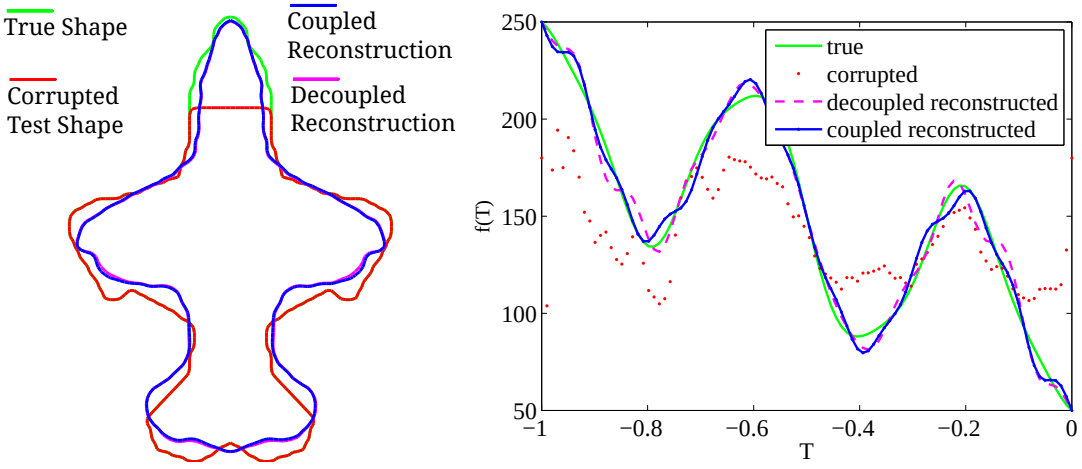
PCA residuals, Figure 33 reveals that TM-coupled outperforms TM-separate in both shape and radiance for heavy noise, which corroborates our assumption that coupled PCA can exploit shape-radiance-correlations to recover objects more robustly in the presence of noise and occlusion.

In conclusion, this experiment demonstrates that relatively similar (Test 2) or even more accurate (Test 1 and Test 3) reconstruction from additive noise and occlusion

(a) noise=10 (b) noise=50 (c) noise=100 (d) noise=150 (e) noise=200 (f) noise=300

Figure 31: Example test image for test 3 with radiance corrupted by varying noise levels and the top 25% of the image occluded.

(a) original image (b) corrupted image (c) reconstructed image (decoupled) (d) reconstructed image (coupled)



(e) shape (f) radiance

Figure 32: Example reconstruction with noise level 300 and 25% occlusion.

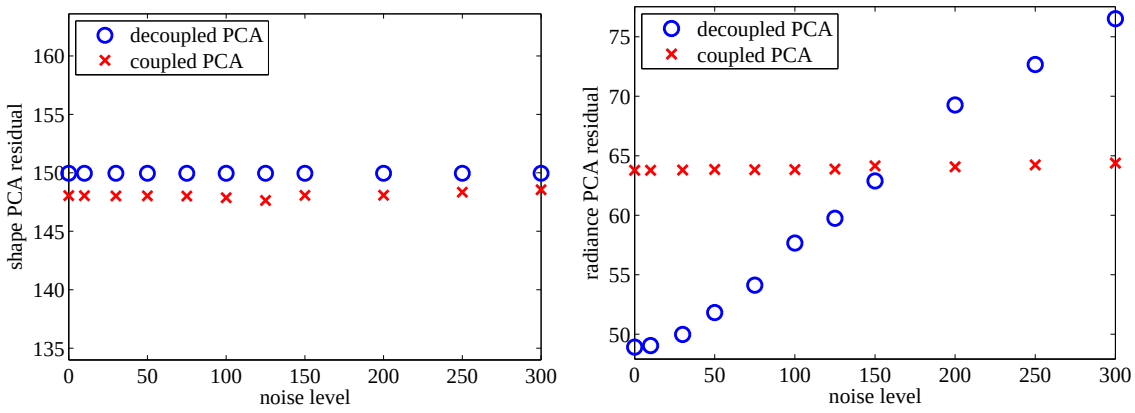


Figure 33: Test 3 result: PCA residuals over noise level when top 25% of the image occluded.

can be obtained with half as many training components by using the coupled shape-plus-radiance model as opposed to separate training of shape and radiance.

3.4 Anisotropic Mumford-Shah Recognition-Segmentation

In this section, anisotropic Mumford-Shah is equipped with prior PCA models as described in Section 3.3, which will help the algorithm localize target objects more robustly if the target shape and radiance information is available. As an introduction, we first review the method in [121], where the Chan-Vese energy (and other low-level energies) are minimized under the constraint that the shape be a particular instance of the shape PCA training model. In this way, the energy is not minimized with respect to an active contour anymore, but with respect to the weight parameters in the PCA training model. Finally, the anisotropic Mumford-Shah functional is reformulated to be restricted to a shape-plus-radiance prior PCA model, which was trained on the coupled shape-plus-radiance vector as explained in Section 3.3.4. Thus, shape and radiance information are jointly exploited which will be shown to mutually benefit each other by providing a richer model of the object compared to a shape with constant radiance in [121].

3.4.1 Introduction: Chan-Vese with Shape Priors

In [121], the Chan-Vese energy is minimized with respect to the weights of the shape PCA prior model from Section 3.3.2. The procedure consists of two steps, the training phase and the recognition phase.

The training phase is described in Section 3.3.2 and consist of aligning training shapes and performing PCA on their signed-distance function vectors. The shape training model is the written as

$$\Phi(x;\boldsymbol{w}) = \bar{\Phi}(x) + \sum_{i=1}^{K} w_i \Phi_i(x). \tag{74}$$

To deal with misalignment between the training shapes and the target shape, a set of pose parameters $\boldsymbol{p}$ (usually translation, rotation, and scale) is introduced by the invertible domain transformation

$$\hat{x} = g(x;\boldsymbol{p}), \tag{75}$$

where $\boldsymbol{p}$ are a finite number of parameters (e.g., translation, rotation, scale), and define

$$\hat{\Phi}(\hat{x};\boldsymbol{w},\boldsymbol{p}) = \Phi(g^{-1}(\hat{x};\boldsymbol{p});\boldsymbol{w}). \tag{76}$$

The transformed $\hat{x}$ variables represent the image domain in the recognition phase, i.e., $I = I(\hat{x})$, whereas the x variables represent the domain where training was performed.

For the segmentation energy, Eq. (76) is assumed to be the shape model and the Chan-Vese energy

$$E_{CV}(\boldsymbol{w},\boldsymbol{p}) = \int_{\hat{R}(\boldsymbol{w},\boldsymbol{p})} (I - u_{in})^2 \, d\hat{x} + \int_{\hat{R}^c(\boldsymbol{w},\boldsymbol{p})} (I - u_{out})^2 \, d\hat{x}, \tag{77}$$

is minimized with respect to the principal component weights $\boldsymbol{w}$ and the pose parameters $\boldsymbol{p}$. Here, $\hat{R}(\boldsymbol{w},\boldsymbol{p})$ is the object domain, i.e.,

$$\hat{R}(\boldsymbol{w},\boldsymbol{p}) = \{x \in \Omega : \hat{\Phi}(\hat{x};\boldsymbol{w},\boldsymbol{p}) < 0\}, \tag{78}$$

and u_{in} (u_{out}) are the object (background) mean intensities, respectively. The resulting gradient descent update equations are given analytically by

$$\boldsymbol{p}_{new} = \boldsymbol{p}_{old} + \Delta t_p(\nabla_{\boldsymbol{p}} E_{CV}), \tag{79a}$$

$$\boldsymbol{w}_{new} = \boldsymbol{w}_{old} + \Delta t_w(\nabla_{\boldsymbol{w}} E_{CV}), \tag{79b}$$

where

$$\nabla_{\boldsymbol{w}} E_{CV} = -2(u_{in}\nabla_{\boldsymbol{w}} S_{in} + u_{out}\nabla_{\boldsymbol{w}} S_{out}) + (u_{in}^2\nabla_{\boldsymbol{w}} A_{in} + u_{out}^2\nabla_{\boldsymbol{w}} A_{out}), \tag{80a}$$

$$\nabla_{\boldsymbol{p}} E_{CV} = -2(u_{in}\nabla_{\boldsymbol{p}} S_{in} + u_{out}\nabla_{\boldsymbol{p}} S_{out}) + (u_{in}^2\nabla_{\boldsymbol{p}} A_{in} + u_{out}^2\nabla_{\boldsymbol{p}} A_{out}), \tag{80b}$$

and

$$\nabla_{w_i} A_{in} = -\nabla_{w_i} A_{out} = -\int_{\hat{C}} \hat{\Phi}_i(\hat{x})\, ds, \tag{81a}$$

$$\nabla_{w_i} S_{in} = -\nabla_{w_i} S_{out} = -\int_{\hat{C}} I\hat{\Phi}_i(\hat{x})\, ds, \tag{81b}$$

$$\nabla_{p_i} A_{in} = -\nabla_{p_i} A_{out} = -\int_{\hat{C}} \nabla_{p_i}\hat{\Phi}(\hat{x})\, ds, \tag{81c}$$

$$\nabla_{p_i} S_{in} = -\nabla_{p_i} S_{out} = -\int_{\hat{C}} I\nabla_{p_i}\hat{\Phi}(\hat{x})\, ds. \tag{81d}$$

The term $\nabla_{p_i}\hat{\Phi}(\hat{x})$ can be computed as

$$\nabla_{p_i}\hat{\Phi}(\hat{x}) = \nabla_{\hat{x}}\hat{\Phi}(\hat{x}) \cdot \frac{\partial g(x;\boldsymbol{p})}{\partial p_i}. \tag{82}$$

To segment an object, an initial guess on the shape component weights $\boldsymbol{w}$ (usually zero, i.e. begin with the mean shape) and the pose $\boldsymbol{p}$ parameters is made, and these gradient descent equations are iterated until convergence.

3.4.2 Anisotropic Mumford-Shah with Shape-Plus-Radiance Priors

A drawback of the Chan-Vese with shape prior is its lack of a radiance training model: While different energies could be considered to be combined with the shape prior model, it is not obvious how specific radiance prior information could be integrated in this approach. This drawback is what the following algorithm addresses. By

constraining both the shape and the radiance to PCA training models, it becomes possible to train on both.

First, consider the decoupled case where shape and radiance are restricted to

$$\Phi(x;\boldsymbol{w}) = \bar{\Phi}(x) + \sum_{i=1}^{K} w_i \Phi_i(x), \tag{83a}$$

$$\mathrm{F}(\tau;\boldsymbol{v}) = \bar{\mathrm{F}}(\tau) + \sum_{i=1}^{K} v_i \mathrm{F}_i(\tau). \tag{83b}$$

Again using pose parameters $\boldsymbol{p}$ (transformed quantities are indicated by $\hat{(\cdot)}$) to allow for shape alignment, we minimize the following aMS energy with shape and radiance priors

$$\begin{aligned} E_{\text{aMS-SR}}(\boldsymbol{w},\boldsymbol{v},\boldsymbol{p}) = &\int_{\hat{R}(\boldsymbol{w},\boldsymbol{p})} \alpha \left(I - \mathrm{F}(\hat{\Phi};\boldsymbol{v})\right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi};\boldsymbol{v}) \|\nabla_{\hat{x}} \hat{\Phi}\|\right)^2 d\hat{x} + \dots \\ &+ \int_{\hat{R}^c(\boldsymbol{w},\boldsymbol{p})} \alpha \left(I - u_{out}\right)^2 d\hat{x}. \end{aligned} \tag{84}$$

Note that the background data term is modeled as a constant (Chan-Vese), since in practice, training data is rarely available for the background, so that we need to resort to a generative model. Also observe that the smoothness term penalizes the derivative of $\mathrm{F}(\tau)$ times the norm of the gradient of the level set function because this is the gradient measure of the radiance in the two-dimensional image domain, as opposed to the one-dimensional domain of the photo-geometric representation. Penalizing the two-dimensional gradient is critical to prevent the level set from arbitrarily shrinking to lower the cost.

The analytical derivatives of the interior parts of the energy with respect to the arguments are derived in detail in Appendix A.3. Here, the result is simply stated as

$$\begin{aligned} \frac{\partial E^{in}_{\text{aMS-SR}}}{\partial p_i} = &\int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0)\right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \|\nabla_{\hat{x}} \hat{\Phi}\|\right)^2\right) \left(\hat{N} \cdot \frac{\partial g}{\partial p_i}\right) d\hat{s} + \dots \\ &+ 2 \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \left(\nabla_{\hat{x}} \hat{\Phi} \cdot \frac{\partial g}{\partial p_i}\right) \left(\alpha \left(I - \mathrm{F}(\hat{\Phi})\right) - \beta \frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \|\nabla_{\hat{x}} \hat{\Phi}\|^2\right) d\hat{x} + \dots \end{aligned} \tag{85}$$

$$- 2\beta \int_{\hat{R}} \left(\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \right)^2 \nabla_{\hat{x}} \hat{\Phi} \cdot \left(H_{\hat{x}} \hat{\Phi} \frac{\partial g}{\partial p_i} + \frac{\partial^2 g}{\partial x \partial p_i} \left[\frac{\partial g}{\partial x} \right]^{-1} \nabla_{\hat{x}} \hat{\Phi} \right) d\hat{x}$$

for the pose parameters, where $H_{\hat{x}}\hat{\Phi}$ denotes the Hessian of $\hat{\Phi}$,

$$\begin{aligned} \frac{\partial E^{in}_{\text{aMS-SR}}}{\partial w_i} = &- \int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0) \right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \|\nabla_{\hat{x}} \hat{\Phi}\| \right)^2 \right) \frac{\hat{\Phi}_i}{\|\nabla_{\hat{x}} \hat{\Phi}\|} d\hat{s} + ... \\ &- 2 \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \hat{\Phi}_i \left(\alpha \left(I - \mathrm{F}(\hat{\Phi}) \right) - \beta \frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \|\nabla_{\hat{x}} \hat{\Phi}\|^2 \right) d\hat{x} + ... \\ &+ 2\beta \int_{\hat{R}} \left(\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \right)^2 \hat{h}_i \cdot \left(\left[\frac{\partial g}{\partial x} \right]^{-1} \nabla_{\hat{x}} \hat{\Phi} \right) d\hat{x}, \end{aligned} \tag{86}$$

for the shape PCA weights, where $h_i(x) = \nabla_x \Phi_i(x)$, and so $\hat{h}_i$ is h_i transformed to the image domain analogous to Eq. (76). Finally,

$$\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial v_i} = 2 \int_{\hat{R}} -\alpha \left(I - \mathrm{F}(\hat{\Phi}) \right) \mathrm{F}_i(\hat{\Phi}) + \beta \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \|\nabla_{\hat{x}} \hat{\Phi}\|^2 \frac{d\mathrm{F}_i}{d\tau}(\hat{\Phi}) \, d\hat{x}, \tag{87}$$

for the radiance PCA weights. These gradients are computed in each iteration and the parameters are updated according to the gradient descent scheme.

This derivation assumes that shape and radiance PCAs were performed independently. For the coupled shape-plus-radiance PCA training model according to Section 3.3.4, the shape PCA and radiance PCA components of the coupled shape-plus-radiance PCA are used instead of the decoupled quantities to compute the derivatives above in exactly the same way. Then, due to the fact that shape and radiance are controlled by the same parameter, Eq. (86) and Eq. (87) are added to form the coupled shape-plus-radiance derivative. Gradient descent is performed on the pose parameters and the coupled shape-plus-radiance PCA weights.

3.4.3 Experiments

In the following sections, results for the aMS with shape-plus-radiance prior are shown and discussed. For convenience, we use the following abbreviations:

- CV: Chan-Vese

- CV-S: Chan-Vese with shape prior (see Section 3.4.1)
- aMS-SRd: Anisotropic Mumford-Shah with decoupled shape-plus-radiance prior (see Section 3.4.2)
- aMS-SRc: Anisotropic Mumford-Shah with coupled shape-plus-radiance prior (see Section 3.4.2)
- aMS-SR: Anisotropic Mumford-Shah with shape-plus-radiance prior (either coupled or decoupled).

In Section 3.4.3.1, two synthetic experiments are shown where these methods are compared to each other for synthetic images. The experiments are designed to clearly demonstrate the potential benefits of aMS-SR algorithms. Section 3.4.3.2 uses an IR image to demonstrate how aMS-SR is more robust in recovering shapes by taking into account the radiance structure of the object. Finally, we show three tracking scenarios on IR videos in Section 3.4.3.3 comparing aMS-SRc, CV-S and a SIFT [93] tracker.

3.4.3.1 Synthetic Experiments

For these experiments, the fighter shapes from Figure 23 are reused to recover a fighter in a test image. To create training data, the fighter shapes are over-laid on top of a gray background and a radiance pattern (radial pattern in Experiment 1, and a leopard pattern in Experiment 2) is assigned to the object region. Then, shape PCA, radiance PCA and shape-plus-radiance PCA are performed on these 12 training images to obtain PCA training models for the CV-S, aMS-SRd, and aMS-SRc algorithms.

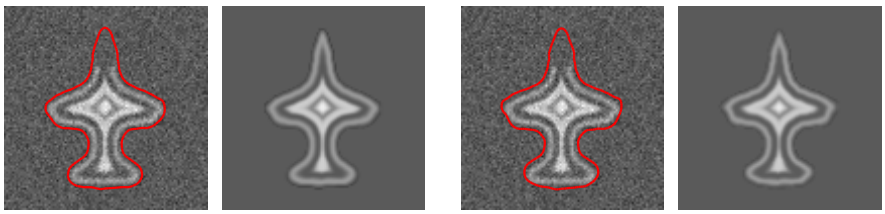
In the test phase, a test image is created from the training data (as is explained below for each experiment), the initial shape and radiance models are set to the mean of the PCA analysis, and pose parameters (translation, scale, rotation) are offset from the true values. Then, CV, CV-S, aMS-SRd, and aMS-SRc are executed to

attempt recovery of the fighter location and shape in the test image.

Synthetic Experiment 1. In the first experiment, the training radiance is a radial pattern similar to the experiment performed in Section 3.3.5. While this might not be a very realistic scenario, it demonstrates well the mechanism of aMS-SR because radial patterns exactly match the aMS-SR model. To make the experiment match the aMS-SR model even better, the test image is created by using the first two principal components of each shape PCA and radiance PCA so that aMS-SR should in theory be able to recover shape, radiance, and pose exactly. However, the recovery task is complicated by adding zero-mean Gaussian noise with variance 15 to the test image and by assigning the top 30% of the image to the background, thus simulating occlusion of the fighter.

The localization results using CV, CV-S, aMS-SRd, and aMS-SRc are shown in Figure 34. Clearly, CV with no shape prior is an over-simplified model for this scenario where there is no clear difference of means between object and background.

(a) initial (b) CV (c) CV-S



(d) aMS-SRd (e) aMS-SRc

Figure 34: Comparison of shape-plus-radiance aMS with Chan-Vese (without and with shape PCA) on fighters with radial patterns. The left image for each method shows the final contour, the right image shows the modeled image.

CV-S performs much better, indicating that shape priors are an important component to add, but CV-S gets stuck in a local minimum. Both aMS-SRd and aMS-SRc recover the shape visually perfectly, as was expected. The energy plot for aMS-SRd and aMS-SRc in Figure 35 reveals two aspects: First, aMS-SRd reaches a smaller energy value, which is expected because the test image was created using the decoupled PCA components, but also because aMS-SRd is the more general model. On the other hand, this generality with twice as many principal components compared to aMS-SRc makes aMS-SRd less robust, as can be seen in the energy plot at iteration 1700, where aMS-SRd barely escapes a local plateau to reach its final global minimum. The aMS-SRc, in contrast, decreases the energy more smoothly.

Synthetic Experiment 2. The same experiment is repeated with a slightly more realistic radiance pattern, which is not radial. To create training images, the same "leopard" texture is added to the fighter shapes. Note that, even though the pattern is the same for all fighter shapes, the photo-geometric representation (radiance) is not

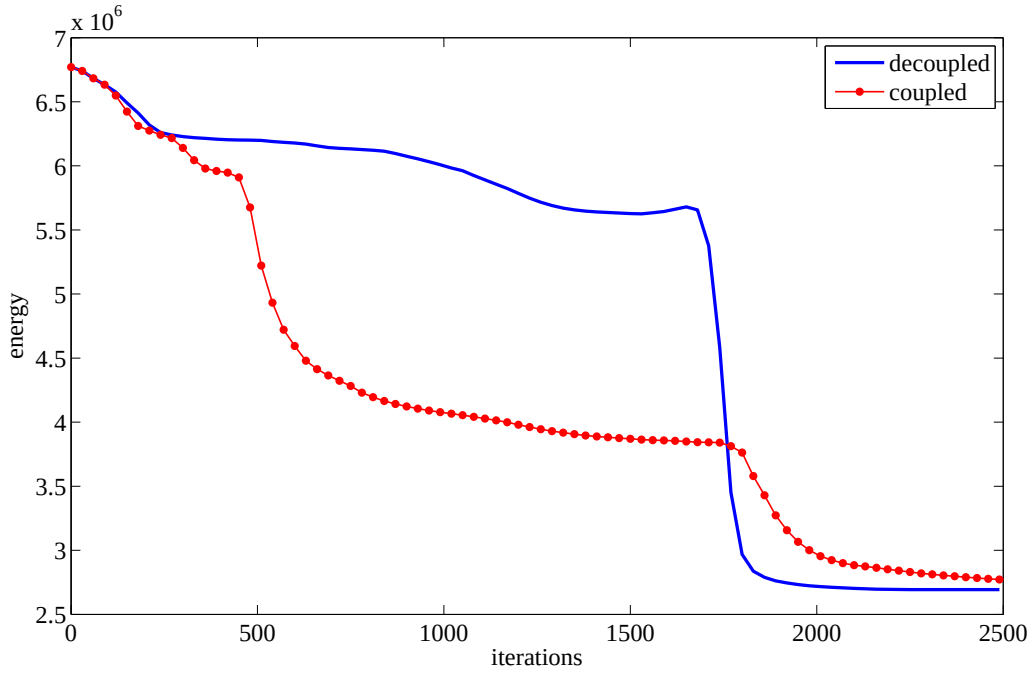


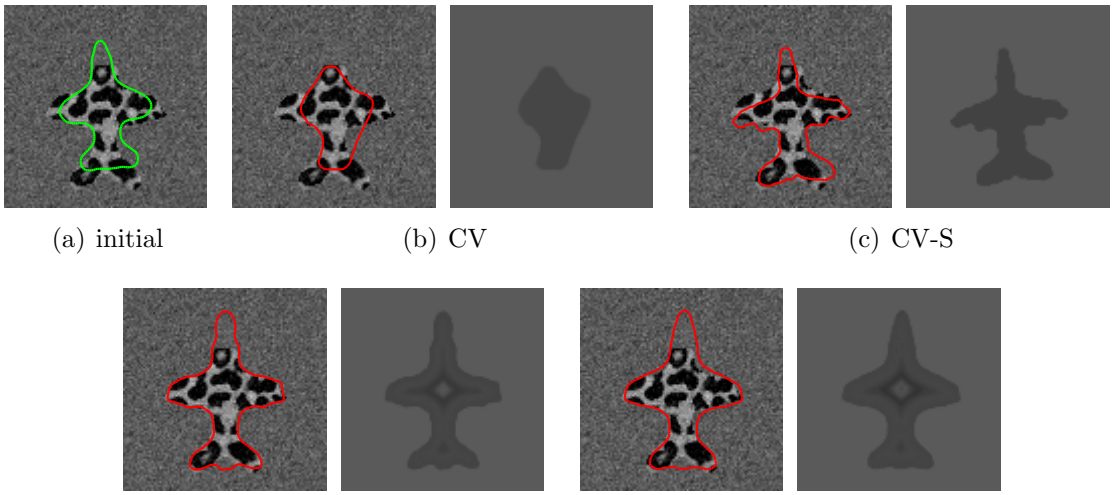
Figure 35: Energy plot for decoupled and coupled shape-plus-radiance aMS for fighters with radial pattern.

the same for the training samples because shape varies between the samples and so does the photo-geometric representation because it depends on shape by definition.

The test image is taken from the training data, but again corrupted by added noise and occlusion. Segmentation results are shown in Figure 36. Again, CV and CV-S do not provide accurate segmentation, whereas aMS-SRd and aMS-SRc are fairly accurate in recovering the global shape and most of its details. It cannot be expected to recover all details of shape since the PCA models use only a certain number of principal components so that, even though the test image is part of the training data, not all of the test image's information is present in the training model, but rather some of its dominant characteristics. The energy plot in Figure 37 reveals again that aMS-SRd is the more general and thus more accurate method. aMS-SRc, however, achieves a similar result with only half of the number of unknown PCA weights.

3.4.3.2 Shape Priors versus Shape-Plus-Radiance Priors

The following experiment further demonstrates superior robustness of aMS-SR compared to CV-S. Consider the infrared image of a beetle from [100] in Figure 38(a)



(a) initial (b) CV (c) CV-S

(d) aMS-SRd (e) aMS-SRc

Figure 36: Comparison of shape-plus-radiance aMS with Chan-Vese (without and with shape PCA) on fighters with leopard pattern. The left image for each method shows the final contour, the right image shows the modeled image.

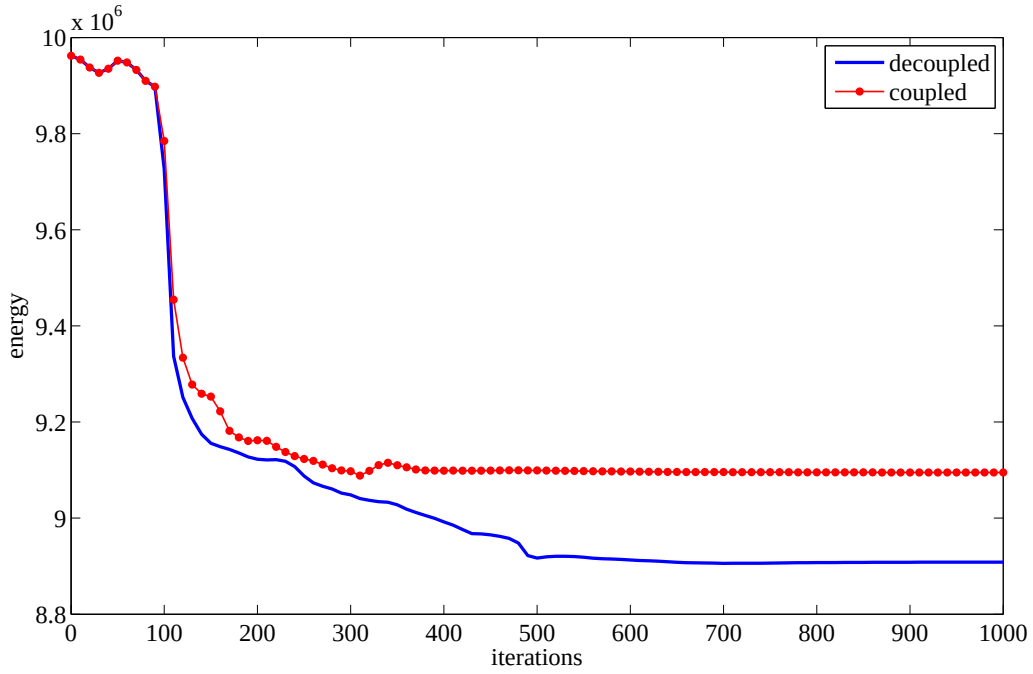
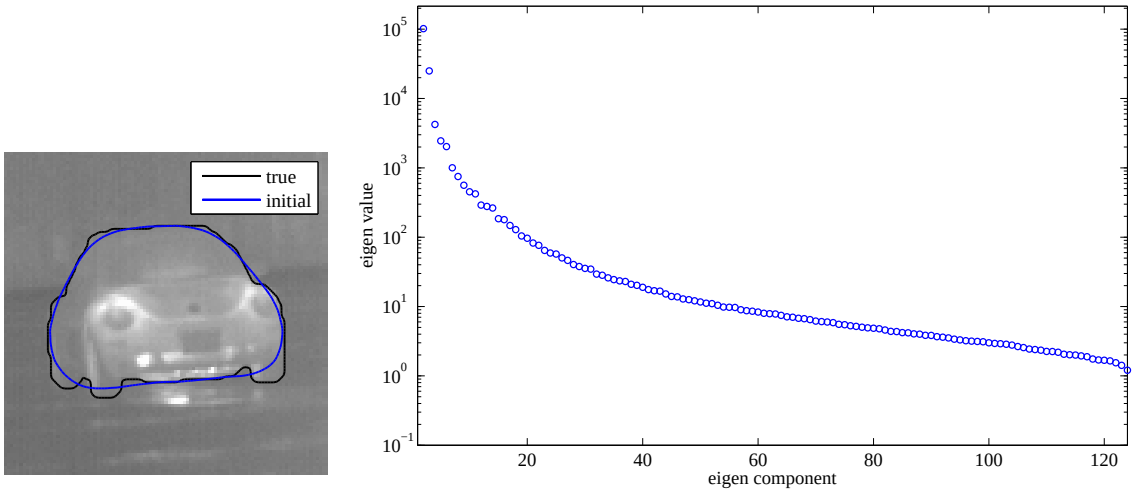


Figure 37: Energy plot for decoupled and coupled shape-plus-radiance aMS for fighters with leopard pattern.

with true segmentation given. A noisy shape model is obtained by shifting and rotating the true shape in different combinations and then performing shape PCA on the dislocated and rotated shapes. The resulting mean shape is used as the initial segmentation guess in Figure 38(a) and the eigenvalues are shown in Figure 38(b) on a logarithmic scale.

In the experiment, we take the radiance model for aMS-SR to be the photo-geometric representation of the original object (this would be equivalent to performing PCA on identical radiance functions), and for CV-S, the true intensity mean of the object is computed and fixed in the CV-S algorithm. Then, K, the number of eigenshapes used in the shape model, is varied and the segmentation results are compared. Note that a larger K is expected to produce more accurate results, since more information about the true object shape will be contained in the training model for larger K.

Visual results are shown in Figure 39 for CV-S in green and aMS-SR in red.

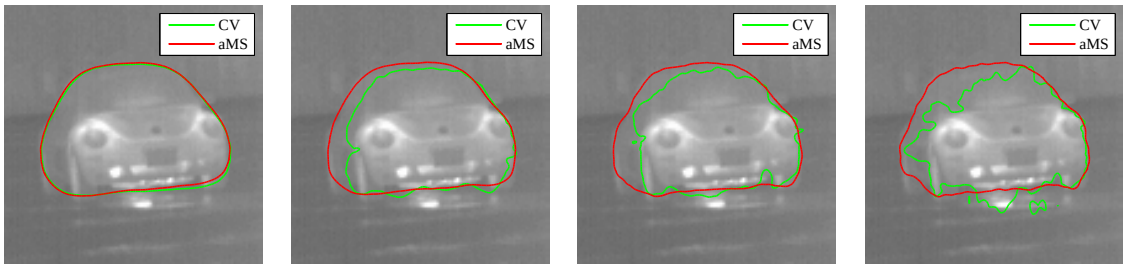


(a) original image (b) eigenvalues of eigenshapes

Figure 38: Original image and training result for beetle experiment.

Whereas for $K = 3$, the results are very similar to the mean shape, the larger K becomes the more CV-S deteriorates and the more accurate aMS-SR becomes. This observation is confirmed in Figure 40(a) where the segmentation error (integral of the square of the difference between the true and estimated object region) is plotted for various values of K. Clearly, aMS-SR tends to decrease segmentation error for larger K, whereas CV-S increases the error and is not robust for $K > 3$.

This result suggests that including more accurate radiance information in aMS-SR compared to CV-S provides a more robust recognition algorithm since regional structure of the radiance component can be exploited. This claim is further corroborated



(a) $K = 3$ (b) $K = 10$ (c) $K = 30$ (d) $K = 60$

Figure 39: Segmentation results for beetle experiment for varying numbers of eigenshape components K.

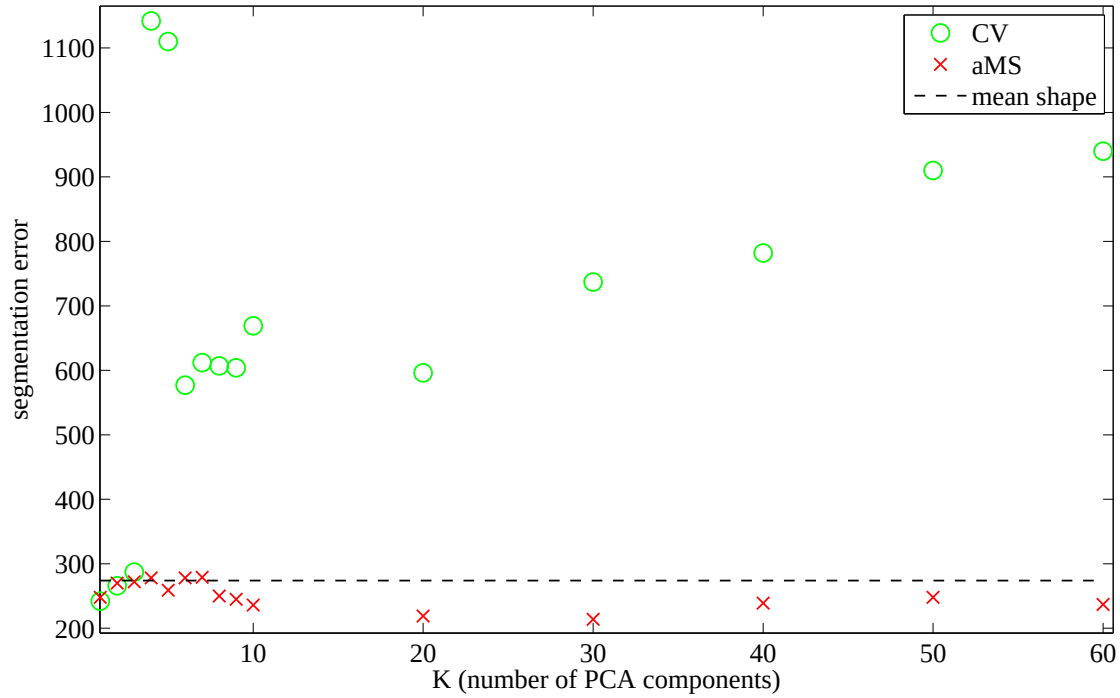
by the plot in Figure 40(b), where the final minimized energy value for a certain K is shown in relation to the final minimized energy value when $K = 1$. It is observed that for CV-S, the final energy value decreases relatively faster for larger K than it does for aMS-SR. This plot indicates that CV-S achieves its goal of minimizing the Chan-Vese energy correctly, but the Chan-Vese energy is not a good model for this example since the actual segmentation error increases. The aMS-SR, on the other hand, does not decrease the final energy values as drastically, but does so to actually approach the global minimum of the ground truth.

3.4.3.3 Tracking Experiments

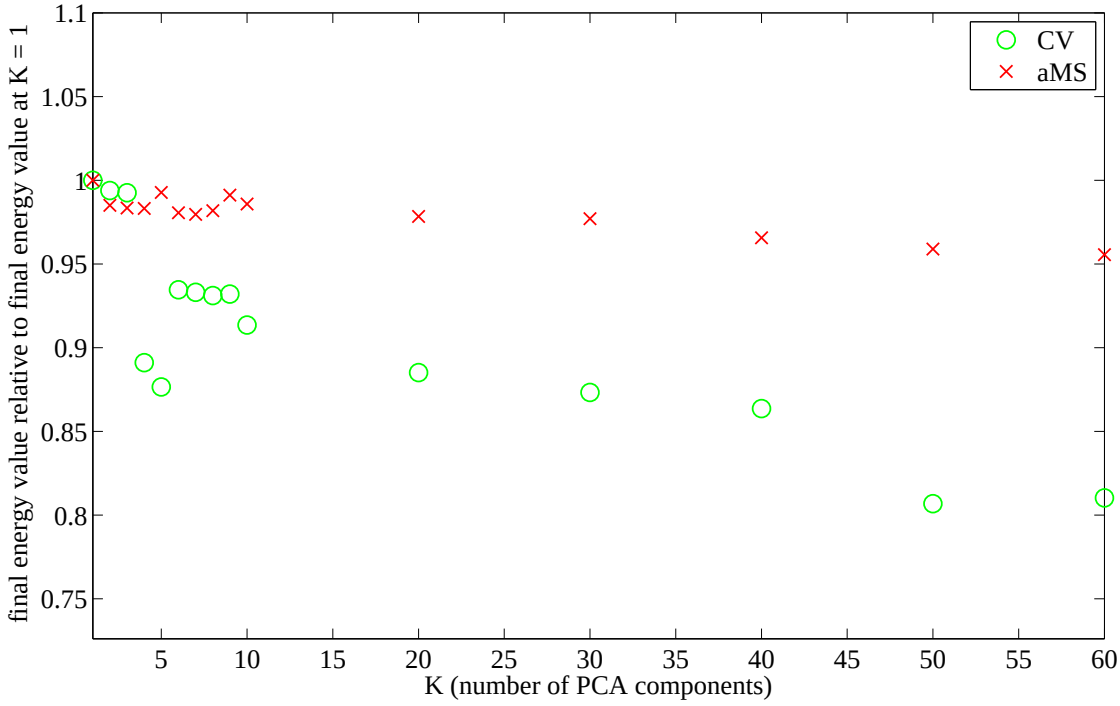
Finally, some tracking results are shown for two videos from the Terravic database [99] and for an additional more challenging online video. The videos are referred to as "irw07", "iruw02 and "truck". We compare tracking results to a tracker using SIFT [93] features implemented in the software VLFeat [124]. The SIFT tracker works as follows:

- compute SIFT descriptors for the object in the initial frame.
- for each frame:
    1. compute SIFT descriptors on the whole image
    2. match (using matching function in VLFeat) these descriptors to the descriptors of the initial frame and to the tracked descriptors of the previous frame. A feature that was successfully matched to the initial or previous frame is "tracked". A descriptor that does not have a matching descriptor in the initial or previous frame is discarded.

For CV-S and aMS-SR algorithms, a shape and shape-plus-radiance PCA model is first learned based on a few images from the sequence. In the tracking phase, the



(a) segmentation error



(b) minimum energy value

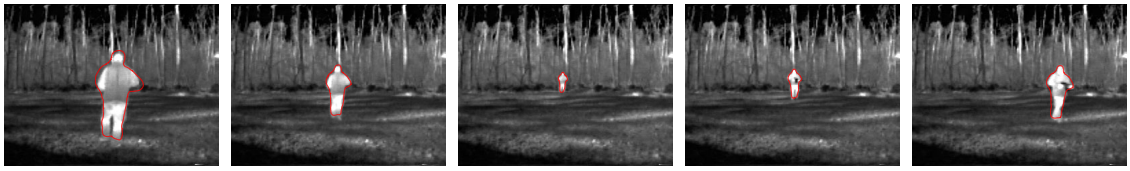
Figure 40: Quantitative results for beetle experiment for varying numbers of eigenshape components K.

algorithms are performed with the initial guess in each frame being the final estimate of the previous frame.

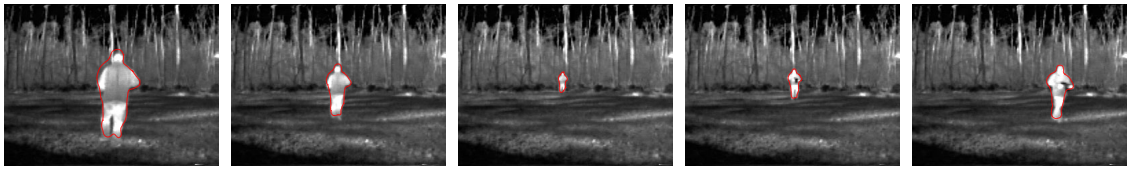
The results are shown in Figure 41 to Figure 43, the first row being aMS-SR, the second row being CV-S and the third row being SIFT, where tracked features are marked.

For irw07 in Figure 41, the CV-S and aMS-SR training models are obtained from 12 frames of various videos from the Terravic database also including three frames from irw07. Since the person to be tracked is a very bright object with high contrast to the background, it is not surprising that CV-S and aMS-SR perform well. SIFT performs well in the first couple of frames. Once the person becomes small and eventually turns around, SIFT features stop tracking due to the lack of salient features and change of pose.

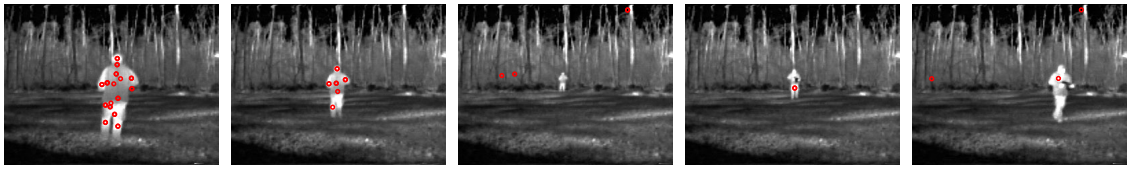
A more challenging scenario is shown for iruw02 in Figure 42, since contrast of the diver's body to the background is quite low. The CV-S and aMS-SR training models are obtained from four frames of this video. aMS-SR performs decently in



(a) aMS-SR



(b) CV-S



(c) SIFT

Figure 41: Tracking results on Terravic video irw07.

tracking the essential characteristics of the object shape. CV-S, on the other hand, tends to spill to the surroundings, which is due to the fact that the object consist of two approximately constant regions (head and body), which is not very well modeled by the Chan-Vese energy. SIFT performs mediocre in this scenario, as most of the time the object is tracked with few features, but quite a few falsely tracked features are observed.

Finally, a video with more radiance structure is shown in Figure 43. Both the truck to be tracked and the background consist of regions with varying intensities, and the truck undergoes a change of pose. Again, CV-S and aMS-SR are trained on a few images of the scene. aMS-SR provides very accurate tracking, whereas CV-S keeps track but spills in the background. This video clearly demonstrates how aMS-SR is more robust than CV-S for an object that has radiance structure and cannot easily be distinguished from the background by only considering means. SIFT tracks well for the first few frames, but then loses track the more the pose of the truck changes.

(a) aMS-SR

(b) CV-S

(c) SIFT

Figure 42: Tracking results on Terravic video iruw02.

(a) aMS-SR

(b) CV-S

(c) SIFT

Figure 43: Tracking results for truck video (Images by courtesy of `http://www.federalinfrared.com/`).

3.5 Conclusion

We have presented two novel segmentation energies based on anisotropic radiance smoothing in the MS functional. Proof-of-concept results are given for potential applications of these energies. The main focus of this chapter, however, has been on shape-plus-radiance priors. The aMS energy gives rise to a one-dimensional photo-geometric representation of an object, which is invariant to translation, scale, and rotation. This representation together with shape in level set formulation is used for PCA training of shape-plus-radiance models, which restrict the evolution of a recognition-segmentation active contour to follow the PCA training model. More robust performance compared to no priors or shape-only priors have been presented for synthetic experiments and real IR data.

Potential future research includes further exploration of aMS without priors, since these formulations were used in this thesis mainly to provide a mathematical connection between the photo-geometric representation and the standard MS functional

through anisotropic smoothing. Especially in medical imaging for radially symmetric objects, aMS is expected to be a viable alternative for existing energies. For recognition tasks with priors, depending on the situation, it may be favorable to consider normalized photo-geometric functions, not its raw values as computed from training images. That is, further improvements can be made by adding an intensity scale factor and offset parameter to the radiance model that allows for stretching the function's range and shifting the mean. These parameters could account for a global change of mean radiance and contrast. This extension would be particularly useful for recognizing the same object in different environments. For tracking, this extension is not crucial as long as the environment does not change too much within the tracking window.

CHAPTER IV

ADJOINT ACTIVE CONTOURS FOR MESHFREE SHAPE-BASED INVERSE SCATTERING PROBLEMS

4.1 Introduction

Level set methods were not invented primarily with the application of image processing in mind, even though they are widely used now in this field. They simply provide a tool for evolving an interface according to a force term. It should not surprise, therefore, that the same level set methods used above for image processing and computer vision can be applied to physical problems where the interface describes the boundary between two regions with different properties, states, phases, etc. The functionals and models for physical problems will be very different from the ones encountered in image processing applications, of course, but the eventual goal of finding an interface and the implementation of interface evolution as level set evolution are shared by these two applications.

In this chapter, we specifically consider shape-based inverse scattering problems for acoustic and elastodynamic probing and image formation. The exact meaning of all these terms will be explained in more detail below, but the overall problem is the following: Assuming there is an ambient domain (e.g., the soil) and an inclusion domain (e.g., an oil reservoir), we launch mechanical (acoustic or elastodynamic) waves in the ambient domain and measure waves scattered from the inclusion and from other boundaries using receivers (sensors) located on the surface of the ambient domain. The goal is then to infer the shape of the inclusion from these scattered wave measurements. Problems of this type are not only relevant to geophysics, but also to medical imaging (ultrasound), and non-destructive testing of structures.

A note on notation in this chapter: Since we will reserve the letter "R" to refer to measurement receivers, we will not use R anymore to refer to a region, but will use Ω^i instead, where Ω^i is the ith region and the whole domain Ω is comprised of the union of all the Ω^i. Moreover, the symbol Γ is used for boundary interfaces to indicate that they are physical interfaces. This notation also makes it clear that both 2D curves and 3D surfaces (not just curves) are relevant for these physical problems, and the proposed methods work analogously for both cases.

4.1.1 Review

The first part of this review will put our striving in the context of the vast field of image formation. Then, methods are reviewed that are more closely related to our research. In particular, inverse scattering and shape-based approaches are discussed in more detail.

Imaging. Image creation has come a long way from depicting or recording visible objects using paint or cameras to modern non-invasive probing in areas such as medical imaging, non-destructive testing, or seismic exploration. Modern imaging techniques can be categorized in different ways: First of all, they employ different physical processes, which carry information about the scene. In many cases, these processes are waves such as light (e.g., photography, optical tomography), infrared waves (e.g., thermography), X-rays (e.g., plain film tomography, computed tomography (CT)), low frequency electromagnetic waves (e.g., radar), acoustic waves (e.g., sonar, ultrasound), or elastic waves (e.g., ultrasonic testing, seismic tomography). Other methods rely on electrical currents (e.g., electrical impedance tomography (EIT)), magnetic fields (e.g., magnetic resonance imaging (MRI)) or static elastic deformation (e.g., elastography). Our proposed research is based on wave processes with a focus on acoustic and elastic waves, but electromagnetic waves are included as well since the mechanics of all waves are fundamentally the same. Most wave-based

imaging techniques exploit either reflection or transmission/absorption phenomena, except, for example, thermography, which is based on the heat radiation of the object itself. Transmission-based methods are most commonly encountered as CT, where the absorption of X-rays in many different directions through the object is measured. All the other above mentioned wave methods are based on the reflection of waves, which is caused by inhomogeneities (e.g., the object to be imaged) in the medium. Except for rare exceptions (e.g., the sunlight in photography or natural seismic activity in seismic imaging), reflection and transmission methods employ sources designed and controlled by the user to excite the wave, and so does our method.

Finally, the common conception of an image being the projection on a two-dimensional plane such as in common photos, thermal images, or simple X-ray scans, is extended to three dimensions (or even four dimensions considering time) in modern imaging methods. In these cases—since we usually cannot measure at any arbitrary point in a three-dimensional domain, especially when the object consists of a solid material—the three-dimensional image has to be inferred from the data measured at remote locations. This inference task is called the *inversion problem*. Whereas the direct problem is usually associated with predicting the measurements given a certain setup, the direction of inference is reversed for the inversion problem. The inversion of measurements in reflection methods is also known as *inverse scattering*.

Whereas most imaging methods yield a pixel image assigning intensities to fixed and discrete pixel locations, our approach falls into a new category known as *shape-based* methods: The target object is modeled as a generic shape in an ambient space, and the goal is to find the shape and material properties of the object, which best match the sensor measurements. Traditionally, this object *segmentation* step (i.e., partitioning the scene into "object of interest" and "background") is performed on the pixel image, i.e., it is an image (post)processing step. In the shape-based approach, the segmentation task is built into the image formation procedure while the formation

of a pixel image is omitted. This approach is advantageous or even necessary when the formation of a pixel image is either computationally expensive or experimentally impractical, for example, when the number of measurements is too small to recover a pixel image with reasonably high resolution. Figure 44 illustrates this difference between traditional pixel-based imaging and shape-based imaging.

Inverse Scattering. Following [81], an inverse problem is defined via its associated direct problem: The inverse problem seeks to recover the data which have led to a certain solution of the direct problem. Whereas the direct problem is usually well-studied and well-posed, the inverse problem is often ill-posed due to lack of uniqueness, existence or stability of the solution. Inverse scattering theory [28, 21] has emerged as one sub-field of inverse problem theory involving the scattering of waves from objects, also called scatterers. In particular, the direct problem is to compute the wave field given excitation sources, material properties and the geometry of the scatterers. The inverse scattering problem, on the other hand, seeks information about the sources, or the scatterers' material properties and geometry from given measurements of the wave field. Early investigations typically made simplifying assumptions such as free space (no boundaries); far-field (measurements far

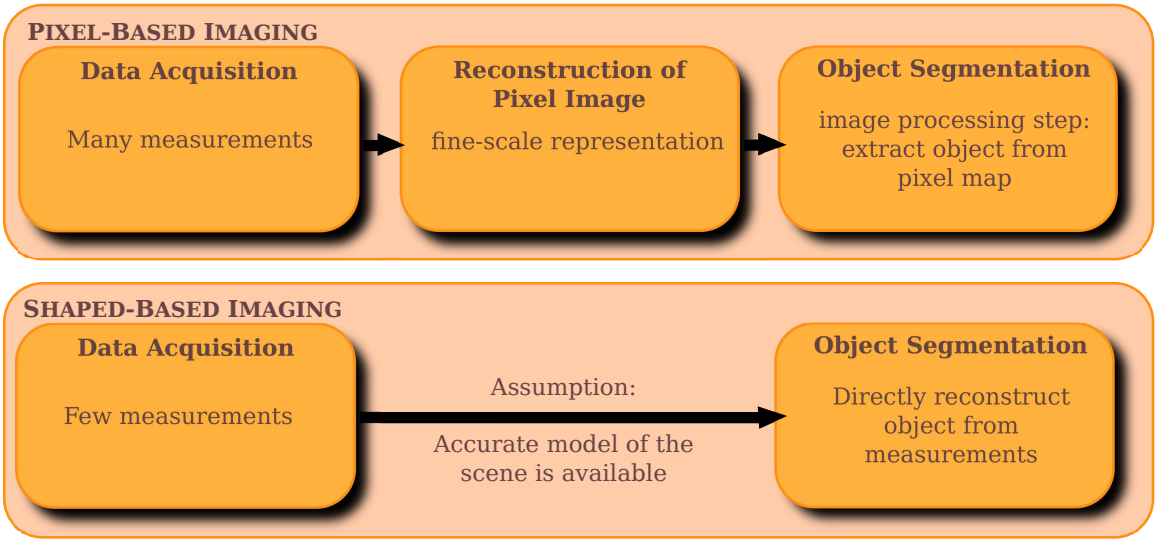


Figure 44: Pixel-based versus shape-based imaging.

away from the scatterer); large number of or even continuous measurement points; and harmonic, scalar, linear wave equation (as opposed to transient, vector-valued or non-linear waves). [26] provides a nice survey for some of the earlier developments, including the linear sampling method [27] and the related factorization method [80], which are based on far-field measurements for all possible incident directions. In [25], it is shown that the MUSIC (Multiple Signal Classification) algorithm is closely related to the linear sampling method. MUSIC is a time reversal method [49], which makes use of the time reversibility of waves, i.e., as opposed to diffusion processes, time inversion for wave processes is a mathematically well-posed problem. Again, many measurements are needed to recover point scatterers [57], but also extended objects [97]. Variational methods are already proposed by [85] (Modified Gradient Method) and later [122] (Contrast Source Method), which minimize a data misfit energy with respect to the scatterer's unknown refraction of index.

In the following, some of the limiting assumptions of the ideal inverse scattering theory presented so far, e.g., far-field or continuous measurements, are dropped at the cost of increased complexity and difficulty of the problem.

Shape-Based Methods. A data misfit energy quantifying the discrepancy between measured and simulated wave amplitudes is the starting point for shape-based methods, whose objective is to find the object shape minimizing the misfit energy. The first-order necessary condition for a minimum puts the concept of shape sensitivity or shape derivative in a central position to these methods. A special case of shape derivative is the topological derivative [105, 12, 47, 58]: Given the object-free continuum, this derivative evaluates at which point in space the insertion of an infinitesimally small object decreases the energy the most. The topological derivative is, therefore, ideal for shape initialization for iterative schemes that reconstruct size and details of the object shape. These shape-based methods iteratively update the estimated shape in the direction of the shape derivative (gradient descent), which

causes the misfit energy to decrease to its minimum.

The shape derivative can be obtained numerically by taking numerical derivatives of the energy with respect to shape parameters [74], which comes at a high computational cost. If the complexity of the problem admits, analytical or semi-analytical shape derivatives can improve speed considerably. Except for few cases, e.g., [111] using a direct derivative, analytical shape derivatives are often derived via the adjoint state method, or *adjoint method* in short, which is also proposed in this research. By way of the adjoint method, the shape derivative is provided semi-analytically in terms of scattering solutions given the current shape guess, meaning in terms of solutions to the direct problem. Early formulations of the adjoint method for acoustics appear in [90, 39]. In these papers, shape is represented as *level sets* [114], whose use for inverse problems was first mentioned in [112]. We pay particular attention to level set methods because they are employed by our proposed algorithm, but parametrization of shapes is common practice, too [48, 130].

Level set methods for inverse problems continue to be an active research field as illustrated by the survey papers [20, 38]. Extending methods to multiple contours [118, 6, 123], fast marching level set methods [41], and applications of the marching cubes method [62] represent some of the newer developments for more powerful and efficient inversion schemes.

Whereas most publications in the literature deal with acoustic or electromagnetic problems, applications of the adjoint method in elastodynamics are investigated in [11, 44, 59, 13]. The use of level sets to represent shape does not seem to be common practice in the elastodynamic community. One reason for the latter may be the fact that popular algorithms for the direct problem, e.g., the finite element method (FEM) [132] or the boundary element method (BEM) [34], require meshing of the object or of its boundary so that the big advantage of level sets being meshfree is rendered futile, as is noted in [64]. For this reason, this research investigates the use

of meshfree numerical solvers, so called Trefftz or collocation methods [82, 89], in combination with level sets to represent shape.

4.1.2 Overview

The remaining two sections in this introduction provide a short tutorial on the wave processes encountered in this thesis, represented by the Helmholtz equation for acoustics in Section 4.1.3 and the Lamé-Navier equation for elastodynamics in Section 4.1.4. The presentation of the research project starts with the problem statement and solution approach in Section 4.2, where assumptions and notation are defined. Then, the solution is presented in Section 4.3 by deriving energy-minimizing equations that comprise the inversion algorithm. Details on the derivations are provided in Appendix B. The inversion algorithm requires numerical solutions to the wave propagation models, which is why a separate section (Section 4.4) is devoted to numerical boundary value problem solvers, in particular our method of choice, the Method of Fundamental Solutions, which is a meshfree collocation method that naturally interacts with the data structure of level sets. Finally, inversion results, key observations, and advanced algorithm improvements are presented in Section 4.5. Section 4.6 concludes the presentation of this project.

4.1.3 Acoustics and the Helmholtz Equation

Many of the results and derivations in this chapter deal with the Helmholtz equation, a classical and thoroughly studied elliptic partial differential equation named after 19th century German physicist Hermann von Helmholtz. The equation for a twice differentiable and complex-valued function $u(x)$ is given as

$$\Delta u(x) + k^2 u(x) = S(x), \quad x \in \mathbb{R}^n, \tag{88}$$

where $k > 0$ and $S(x)$ is a given forcing function. The Helmholtz equation has great significance in engineering and physics as being the scalar wave equation transformed

to the frequency domain. As such, it models scalar wave processes, for example acoustics. Even though the derivations and results of this doctoral research are fundamental results that apply to the Helmholtz equation in general, we choose to interpret it and refer to it as acoustics because we later expand the results to elastodynamic waves. Both acoustic and elastodynamic waves are mechanical waves, and acoustics can be viewed as a special case of elastodynamics where there is no shear force (e.g., in liquids or gases), and therefore, only longitudinal waves can propagate in the medium, making it a scalar problem. However, this research is more general and fundamental than being only applicable to acoustics. Any wave process that can be modeled as a scalar, linear wave equation, fits the assumptions of this research.

We now show the relationship between the Helmholtz equation and the scalar wave equation. Given the scalar wave equation

$$\Delta u(x,t) - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}u(x,t) = S(x,t), \quad x \in \mathbb{R}^n,\, t > 0, \tag{89}$$

where $c > 0$, and assuming harmonic functions of the form

$$u(x,t) = \hat{u}(x)e^{-i\omega t}, \quad S(x,t) = \hat{S}(x)e^{-i\omega t} \tag{90}$$

for $\omega \geq 0$, Eq. (89) becomes

$$\Delta\hat{u}(x)e^{-i\omega t} - \frac{(-i\omega)^2}{c^2}\hat{u}(x)e^{-i\omega t} = \hat{S}(x)e^{-i\omega t}, \quad x \in \mathbb{R}^n,\, t > 0, \tag{91}$$

leaving us with

$$\Delta\hat{u}(x) + \frac{\omega^2}{c^2}\hat{u}(x) = \hat{S}(x), \quad x \in \mathbb{R}^n \tag{92}$$

after canceling out the exponential terms. Now, defining $k = \omega/c$, the Helmholtz equation in form of Eq. (88) is obtained. The physical interpretation of this procedure is essentially given by Fourier analysis: The signals of the wave process are decomposed into their frequency components and each frequency component is governed by the Helmholtz equation with the constant $k = \omega/c$ depending on the angular

frequency ω of the source and the wave propagation speed c, which is a material property of the medium. The constant k is called wave number and is related to the wave length λ by $k = 2\pi/\lambda$.

Often, when dealing with practical problems, space cannot be modeled as free space but has boundaries on which certain boundary conditions hold. If a domain Ω with differentiable boundary $\partial\Omega = \Gamma$ is considered, many practical problems can be modeled by the following boundary value problem (BVP)

$$\Delta u(x) + k^2 u(x) = S(x), \quad x \in \Omega \tag{93a}$$

$$\gamma u(x) + (1-\gamma)\frac{\partial u(x)}{\partial \mathcal{N}} = 0, \quad x \in \Gamma \tag{93b}$$

where $0 < \gamma < 1$ defines a Robin boundary condition and $\partial u/\partial\mathcal{N} = \nabla u \cdot \mathcal{N}$ is the directional derivative of u in the outward normal direction of the domain boundary. Setting $\gamma = 1$ yields a Dirichlet boundary condition, and $\gamma = 0$ a Neumann boundary condition. In general, analytical solutions to this boundary value problem are hard to obtain and possible only for simple geometries of the domain Ω, e. g. for a sphere, an ellipsoid, or a half space. For this reason, numerical approaches will be employed for approximate solutions to Eq. (93).

4.1.4 Elastodynamics

Elastodynamics describes the propagation of mechanical waves in solids, and is characterized by non-zero shear forces compared to acoustics. This property allows for the propagation of transversal waves in addition to only longitudinal waves in acoustics. This difference is due to the material properties of the three states of matter: Whereas solids resist being bent, liquids and gases do not, because of their lack of inter-molecular bindings. Acoustics can, therefore, be viewed as a special case of elastodynamics where shear forces cannot be transmitted. Since we do not expect the typical reader in Electrical and Computer Engineering to be familiar with elastodynamics, a very short introduction to the definitions and equations is given, also

clarifying our use of notation. Those equations are classical and are found in any textbook on the topic, e.g., [56].

Consider an undeformed, quiescent, continuous, solid object in n-dimensional space. For practical problems, $n = 2$ and $n = 3$ are of interest, but the equations can be written independent of the dimensionality of the problem. Initially, every point $x \in \mathbb{R}^n$ of the object is at rest until a disturbance at time $t_0 = 0$ causes matter to move. The displacement vector $u(x,t) \in \mathbb{R}^n$ describes how far and in which direction a point, which was at x at $t_0 = 0$, has moved at some given time $t \geq 0$. That is,

$$u(x,t) = \tilde{x}(x,t) - x, \tag{94}$$

where $\tilde{x}(x,t) \in \mathbb{R}^n$ is the current location of the point, which was initially at x. The displacement field fully characterizes the kinematic state of the object. This kinematic state includes rigid motion, e.g., when the displacement field is constant everywhere at some time t, and deformational motion, which is caused by neighboring points being displaced in different directions or at different magnitude. It is the latter that elasticity is concerned with. Deformations are characterized by the strain tensor $\epsilon(x,t) \in \mathbb{R}^{n\times n}$, which—under the assumption of small displacements—is defined as

$$\epsilon(x,t) = \frac{1}{2}\left(\nabla u(x,t) + (\nabla u(x,t))^T\right), \tag{95}$$

where ∇ is the spatial gradient and $(\cdot)^T$ indicates the transpose of a matrix. At this point, it becomes convenient to drop (x,t) dependencies and to introduce index notation: A vector will be represented by a subindex, e.g., $u \mathrel{\widehat{=}} u_i$, where $i = 1, ..., n$. The spatial gradient is written as a comma-separated subscript, e.g., $\nabla u \mathrel{\widehat{=}} u_{i,j}$, so $\epsilon \mathrel{\widehat{=}} \epsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i})$ is a tensor with n^2 components. Moreover, it is convention to sum from 1 to n over two identical indices, e.g., $u_{i,i} = u_{k,k} := \sum_{k=1}^{n} u_{k,k} = \text{trace}(\nabla u)$. Note that we refrain from using index notation on spatial vector variables, such as x or ξ, to avoid unnecessary clutter.

The goal is to obtain the equations of motion for the displacement field. According to Newton's laws, motion and force are closely interconnected, and it is thus necessary to define internal forces in the continuum. The average force acting on an infinitesimally small surface with unit normal vector $\mathcal{N}_i$ at a point is called the traction vector $t_i \in \mathbb{R}^n$ at that point in the direction $\mathcal{N}_i$. The state of forces at a point is fully described by the stress tensor T_{ij}, whose jth columns is the traction vector acting on the plane with normal direction being the jth Euclidean basis vector. Then, traction in an arbitrary direction $\mathcal{N}_i$ can be written as the product of T_{ij} with $\mathcal{N}_i$, that is

$$t_i = T_{ij}\mathcal{N}_j. \tag{96}$$

Again, note that by convention, $T_{ij}\mathcal{N}_j = \sum_{j=1}^{n} T_{ij}\mathcal{N}_j$. Moreover, applying conservation of angular momentum shows that $T_{ij} = T_{ji}$, i.e., the stress tensor is symmetric independent of the material.

By introducing the relationship between stress and strain, we now arrive at the point where the terminology "elasticity" becomes clear. In common language, a rubber band or trampoline, which is supported by springs, would be associated with the adjective elastic. In fact, elasticity can be intuitively described as a material property that leads to increasing resisting forces the more the material is stretched. This behavior is modeled by the constitutive equation for a linear elastic isotropic medium, an extension of Hook's Law for springs, as

$$T_{ij} = \lambda\epsilon_{kk}\delta_{ij} + 2\mu\epsilon_{ij}. \tag{97}$$

Here, λ and μ are material specific constants known as the Lamé constants, $\epsilon_{kk} = \text{trace}(\epsilon)$ and δ_{ij} is the Kronecker delta, which is $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$, and thus corresponds to the identity matrix in $\mathbb{R}^n$. In general, both compressional and shear forces are generated by this material law, which makes the behavior of elastic solids fundamentally different from that of fluids and gases, where only compressional

forces are present.

Finally, Newton's law of conservation of momentum will yield the relationship between stresses and the change of motion. Let $\dot{(\cdot)}$ denote partial time derivative and ρ the density of the medium. The conservation of momentum for a volume Ω with surface Γ in the absence of external forces states that

$$\frac{d}{dt}\int_{\Omega} \rho \dot{u}_i \, d\Omega = \int_{\Gamma} T_{ij}\mathcal{N}_j d\Gamma, \tag{98}$$

which after applying the divergence theorem yields

$$\int_{\Omega} \rho \ddot{u}_i - T_{ij,j} \, d\Omega = 0. \tag{99}$$

Since Eq. (99) needs to hold for all possible volumes Ω, the integrand must be zero. Thus,

$$T_{ij,j} = \rho \ddot{u}_i. \tag{100}$$

The second time derivative in Eq. (100) motivates the term "elastodynamics", whereas in static linear elasticity there is no motion and the right hand side of Eq. (100) is zero.

Now, substituting Eq. (97) and Eq. (95) in Eq. (100), we finally arrive at the equation of motion for elastodynamics, the so-called Lamé-Navier equation

$$(\lambda + \mu)u_{j,ji} + \mu u_{i,jj} = \rho \ddot{u}_i, \tag{101}$$

which is a system of n coupled linear partial differential equations. Given well-behaved initial and boundary conditions, uniqueness of solutions can be shown [56].

Often, it is more convenient to analyze the equation in the Fourier domain. The same calculation as in Section 4.1.3 for the scalar wave equation leads to

$$(\lambda + \mu)\hat{u}_{j,ji} + \mu \hat{u}_{i,jj} + \omega^2 \rho \hat{u}_i = 0, \tag{102}$$

where $\hat{u}_i$ is the complex, spatially varying displacement amplitude of the time-harmonic wave field for frequency ω. From now on, frequency domain is assumed and the hat is dropped from variables in the Fourier domain for notational convenience.

4.2 Problem Statement and Solution Approach

4.2.1 Problem Setup

Consider the setup in Figure 45, where a bounded domain Ω^0 is occupied by a homogeneous material. On the outer surface $\bar{\Gamma}$, a source S, e.g., a shaker, excites harmonic waves at frequency ω, and receivers r_i measure the scattering response. Inside Ω^0, one or more inclusions Ω^l, $l = 1, ..., L$ are assumed, which are occupied by a homogeneous material different from Ω^0. We denote the boundaries as $\partial\Omega^l = \Gamma^l$, $l = 0, ..., L$ and the respective outward normal vectors as $\mathcal{N}^l$. Observe that $\Gamma^0 = \bar{\Gamma} \cup \bigcup_{l=1}^{L} \Gamma^l$. Also, we see that the whole domain Ω is the union of the disjoint sets Ω^l, $l = 0, ..., L$.

For our purposes, we assume that we are given the receiver locations and measurements, the source locations and characteristics (e.g., amplitude, frequency, direction of excitation), and domain material properties for $\Omega^0, ..., \Omega^L$. In addition, we assume that the acoustic (for liquids and gases) or elastodynamic (for solids) equations of motion sufficiently describe wave propagation in these domains, and that boundary conditions on the outer surface $\bar{\Gamma}$ are given for the wave fields, e.g. "zero traction" for a free solid surface. The goal of this chapter is to find the boundary shapes

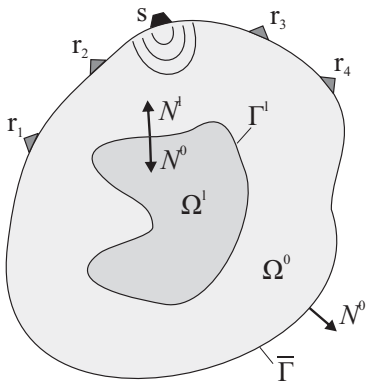


Figure 45: The problem setup: A source S, e.g., a shaker, introduces harmonic mechanical waves into a bounded domain Ω^0. Embedded in Ω^0 are one or more inclusions Ω^l, $l = 1, ..., L$ with different acoustic properties than Ω^0. Using the measurements at the receivers r_i, the goal is to estimate the shape and location of the domains Ω^l, or equivalently, of their boundaries Γ^l.

Γ^l, $l = 1, ..., L$—note that $\bar{\Gamma}$ is fixed and assumed to be known—from this given information.

4.2.2 Energy Definition

To find these boundary shapes, the adjoint active contours problem is formulated to minimize an energy, which assigns a non-negative, real number to a given set of boundaries $\Gamma^1, ..., \Gamma^L$ as follows

$$E(\Gamma^1, ..., \Gamma^L) = \frac{1}{2} \sum_{r=1}^{R} \|u(x_r) - \hat{u}(x_r)\|^2. \tag{103}$$

This functional quantifies how well (or more precisely the opposite: how "bad") a set of shapes describes the given measurements. Here, R is the number of receivers, $x_r \in \bar{\Gamma}$ are the receiver locations, $\hat{u}$ the measured displacement values, and u simulated displacements for given shapes $\Gamma^1, ..., \Gamma^L$. The term "simulated" indicates that there ought be a model for the displacement field u, which can be simulated. This model is provided by the acoustic or elastodynamic (depending on the problem) equations of motion introduced in Section 4.1. These models are partial differential equations with boundary conditions describing the wave mechanics. Given further information about source locations, amplitudes and frequency, a forward solution u can be obtained by solving the boundary value problem. The energy value may then be computed by comparing the simulated amplitudes at receiver locations to the actual measured amplitudes according to Eq. (103). It is obvious that the misfit energy E will provide meaningful results—in the sense that small values for E correspond to reasonable approximations of the boundary shapes—only if the model simulating the wave field u accurately reflects the physics of the problem.

In the following section, the gradient descent minimization procedure for this energy is presented, which results in active contour flow equations that drive an initial guess for the shapes $\Gamma^1, ..., \Gamma^L$ to a minimum of the energy E.

4.3 Energy Minimization

4.3.1 Analytical Shape Derivatives via the Adjoint Method

Letting shapes depend on artificial time t, we can formally apply the chain rule and write the energy variation with respect to t as

$$\frac{dE}{dt} = \sum_{l=1}^{L} \left\langle \frac{\partial E}{\partial \Gamma^l}, \frac{d\Gamma^l}{dt} \right\rangle, \tag{104}$$

where $\langle \cdot, \cdot \rangle$ indicates an inner product and $\frac{\partial E}{\partial \Gamma^l}$ is the *shape derivative* of E with respect to the shape Γ^l. Assuming for a moment that the shape derivative is known, we may set

$$\frac{d\Gamma^l}{dt} = -\frac{\partial E}{\partial \Gamma^l}, \quad l = 1, \ldots, L \tag{105}$$

in order to drive the energy to its minimum over time, since the derivative

$$\frac{dE}{dt} = -\sum_{l=1}^{L} \left\langle \frac{\partial E}{\partial \Gamma^l}, \frac{\partial E}{\partial \Gamma^l} \right\rangle = -\sum_{l=1}^{L} \left\| \frac{\partial E}{\partial \Gamma^l} \right\|^2 \tag{106}$$

will always be non-positive. Eq. (105) is a curve evolution equation and may be implemented like a standard computer vision active contour using level sets. However, the main theoretical task left is to find the analytical shape derivative.

Next, the derivation for acoustics as presented in [29] is reviewed and a few simplifying manipulations are made. Then, the more involved result for elastodynamics is presented. Both derivations for analytical shape derivatives are based on the problem setup from Section 4.2.1.

4.3.1.1 Shape Derivative for Acoustics

A general forward model for the wave field $u : \Omega \mapsto \mathbb{R}$ for acoustics in the frequency domain is given by the equations

$$\mathcal{L}^0 u := -d_0 \Delta u^0 + k_0^2 u^0 = S, \quad \text{in } \Omega^0 \tag{107a}$$

$$\mathcal{L}^l u := -d_l \Delta u^l + k_l^2 u^l = 0, \quad \text{in } \Omega^l, l = 1, \ldots, L. \tag{107b}$$

with boundary conditions (Robin boundary condition on the outer boundary plus two continuity conditions on the interfaces)

$$\lambda u^0 + (1-\lambda)\frac{\partial u^0}{\partial \mathcal{N}^0} = 0, \quad \text{on } \bar{\Gamma}, \tag{107c}$$

$$u^0 - u^l = 0, \quad \text{on } \Gamma^l,\ l = 1, \ldots, L, \tag{107d}$$

$$d_0\frac{\partial u^0}{\partial \mathcal{N}^0} + d_l\frac{\partial u^l}{\partial \mathcal{N}^l} = 0, \quad \text{on } \Gamma^l,\ l = 1, \ldots, L. \tag{107e}$$

Here, for $i = 0, ..., L$, u^i is the displacement wave field u restricted to Ω^i, $k_i = \omega/c_i$ is the wave number (quotient of circular frequency and wave speed) and d_i is a material constant that is useful in electromagnetic problems and will, therefore, be kept for generality. For acoustics, we then simply set $d_0 = d_1 = -1$ to obtain the Helmholtz equation. λ is a weight between 0 and 1 to define Robin boundary conditions and not to be confused with the wavelength.

In the following, the equations minimizing the energy Eq. (103) with the constraint that Eq. (107) holds will be presented. The derivation follows [29]. If we define

$$g(x) = \frac{1}{2}\sum_{r=1}^{R} \|u(x_r) - \hat{u}(x_r)\|^2 \delta(x - x_r), \tag{108}$$

the misfit energy can be rewritten in integral form as

$$E = \int_{\bar{\Gamma}} g(x)\, d\Gamma + \sum_{l=0}^{L} \int_{\Omega^l} (\mathcal{L}^l u - S^l) w^l \, d\Omega, \tag{109}$$

where $S^0 = S$ and $S^l = 0$ for $l = 1, \ldots, L$, so that the second term is zero according to Eq. (107). w^l is an arbitrary function serving as Lagrange multiplier for the PDE-constrained minimization problem and is to be determined later.

The key idea of the minimization procedure is to let the shapes $\Gamma^1, ..., \Gamma^L$ depend on an artificial time t and to derive time-evolution equations for these shapes, which decrease the misfit energy. Note that t cannot be confused with physical time since we consider acoustics in the frequency domain, not in the time domain. Hence, both E and the shapes $\Gamma^1, ..., \Gamma^L$ depend on t. Now, taking the derivative of Eq. (109) via

the adjoint method, one arrives at

$$\frac{dE}{dt} = \sum_{l=1}^{L} \int_{\Gamma^l} \left((d^l - d^0) \nabla u^l \cdot \nabla w^0 + (k^l - k^0) w^l u^l \right) \left(\dot{\Gamma}^l \cdot \mathcal{N}^l \right) \, d\Gamma, \tag{110}$$

where w is derived to be the *adjoint* displacement field satisfying

$$\mathcal{L}^l w = 0, \quad \text{in } \Omega^l, l = 0, \ldots, L, \tag{111a}$$

with boundary conditions

$$d_0 (\lambda w^0 + (1 - \lambda) \frac{\partial w^0}{\partial \mathcal{N}^0}) = -\frac{dg}{du}, \quad \text{on } \bar{\Gamma}, \tag{111b}$$

$$w^0 - w^l = 0, \quad \text{on } \Gamma^l, \, l = 1, \ldots, L, \tag{111c}$$

$$d_0 \frac{\partial w^0}{\partial \mathcal{N}^0} + d_l \frac{\partial w^l}{\partial \mathcal{N}^l} = 0, \quad \text{on } \Gamma^l, \, l = 1, \ldots, L, \tag{111d}$$

where

$$\frac{dg}{du} = \sum_{r=1}^{R} \left(u(x_r) - \hat{u}(x_r) \right) \delta(x - x_r). \tag{112}$$

Also note that we use the notations $\dot{\Gamma}^l = \Gamma^l_t = \frac{\partial \Gamma^l}{\partial t}$ equivalently. The adjoint field w satisfies very similar equations as the *primary* field u in Eq. (107), except that w satisfies a homogeneous PDE with boundary conditions depending on u. The adjoint solution becomes zero if (and only if) the misfit energy becomes zero, because then $\frac{dg}{du} \equiv 0$, and so $w \equiv 0$ satisifies Eq. (111).

Now, according to Eq. (110), define

$$F^l = (d^0 - d^l) \nabla u^l \cdot \nabla w^0 + (k^0 - k^l) w^l u^l, \tag{113}$$

and pick

$$\dot{\Gamma}^l = F^l \mathcal{N}^l. \tag{114}$$

This choice of $\dot{\Gamma}^l$ makes Eq. (110) as negative as possible, and thus decreases the misfit energy as quickly as possible over time. Note that Eq. (113) is a simplified version of the expression in [29]. The simplifications to arrive at this result are presented in Appendix B.1.

Another comment is related to the wave fields u and w being complex-valued. In the derivation of the shape derivative, it is assumed that the wave fields are real. This seeming conflict is resolved by realizing that the shape derivative for complex wave fields can be split into two parts, the shape derivative for the real part u_{Re} plus the shape derivative for the imaginary part u_{Im}. It is clear that both u_{Re} and u_{Im} satisfy the PDEs because of linearity. Also, the energy can be divided according to

$$\begin{aligned} E &= \frac{1}{2}\sum_{r=1}^{R} \|u(x_r) - \hat{u}(x_r)\|^2 \\ &= \frac{1}{2}\sum_{r=1}^{R} \|u_{\mathrm{Re}}(x_r) - \hat{u}_{\mathrm{Re}}(x_r)\|^2 + \frac{1}{2}\sum_{r=1}^{R} \|u_{\mathrm{Im}}(x_r) - \hat{u}_{\mathrm{Im}}(x_r)\|^2, \end{aligned} \tag{115}$$

when u and $\hat{u}$ are complex. Above, both energy components depending only on real and only on imaginary parts, respectively, can be minimized using the adjoint method. Note that the energy is extended in these case by the real (imaginary) part of the adjoint field times the real (imaginary) part of the PDE. The final result for the shape derivative is, therefore,

$$F^l = F^l_{\mathrm{Re}} + F^l_{\mathrm{Im}}, \tag{116}$$

where F^l_{Re} and F^l_{Im} are the shape derivatives resulting from the real and the imaginary parts, respectively.

4.3.1.2 Shape Derivative for Elastodynamics

A similar derivation is now performed for the setup in Figure 45 assuming that the wave process is modeled by elastodynamics. As in the case of acoustics, we only consider real-valued displacements, and the extension to complex-valued fields is obtained as explained above. For elastodynamics, displacement is vector-valued (indicated by the subscript i for u_i), and so the objective is to minimize

$$E(\Gamma^1, ..., \Gamma^L) = \frac{1}{2}\sum_{r=1}^{R} \|u_i(x_r) - \hat{u}_i(x_r)\|^2, \tag{117}$$

where the displacement field u_i satisfies the boundary value problem

$$T_{ij,j}^{(u^l)} + \gamma^l u_i^l = S_i^l \quad \text{in } \Omega^l,\ l = 0, ..., L, \tag{118a}$$

$$T_{ij}^{(u^0)} \mathcal{N}_j^0 = 0 \quad \text{on } \bar{\Gamma}, \tag{118b}$$

$$u_i^l - u_i^0 = 0 \quad \text{on } \Gamma^l,\ l = 1, ..., L, \tag{118c}$$

$$(T_{ij}^{(u^l)} - T_{ij}^{(u^0)}) \mathcal{N}_j^l = 0 \quad \text{on } \Gamma^l,\ l = 1, ..., L. \tag{118d}$$

In words, the wave field satisfies the elastodynamic PDE within each domain, is traction-free on the surface boundary, and has continuous displacement and traction values across inclusion interfaces. In Eq. (118), u_i^l is u_i restricted to Ω^l, $\gamma^l = \rho^l \omega^2$ where ρ^l is density and ω is the angular frequency of the wave process. S_i^l is a given source term in the frequency domain, and $\mathcal{N}_i^l$ is the outward normal vector for Ω^l. Recall that the relation between stress and displacement is stated as

$$T_{ij}^{(u^l)} = \lambda^l u_{k,k}^l \delta_{ij} + \mu^l (u_{i,j}^l + u_{j,i}^l), \tag{119}$$

where λ^l is Lamé's first parameter and μ^l is Lamé's second parameter, also known as shear modulus, and therefore, Eq. (118a) can be written in terms of displacements as

$$(\lambda^l + \mu^l) u_{k,ki}^l + \mu^l u_{i,kk}^l + \gamma^l u_i^l = S_i^l \quad \text{in } \Omega^l,\ l = 0, ..., L, \tag{120}$$

which is the Lamé-Navier equation introduced in Section 4.1.4.

In Appendix B.2, the minimizing equations are derived analogous to acoustics using the adjoint method. For elastodynamics, the calculus becomes more involved because displacement fields are vector-valued. A similar derivation using the adjoint method for layers in two dimensions was presented in [8]. By using index notation in our thesis, however, the problem becomes tractable for arbitrary dimensions. The derivation shows that the curve evolution is again written as

$$\dot{\Gamma}_i^l = F^l \mathcal{N}_i^l, \tag{121}$$

where the force term F^l for elastodynamics is

$$F^l = (T_{ij}^{(u^l)} - T_{ij}^{(u^0)})w_{i,j}^l + (u_{i,j}^0 - u_{i,j}^l)T_{ij}^{(w^0)} + (\gamma^0 - \gamma^l)u_i^l w_i^l, \tag{122}$$

or equivalently

$$F^l = (T_{ij}^{(w^l)} - T_{ij}^{(w^0)})u_{i,j}^l + (w_{i,j}^0 - w_{i,j}^l)T_{ij}^{(u^0)} + (\gamma^0 - \gamma^l)u_i^l w_i^l. \tag{123}$$

The adjoint displacement field w^i satisfies

$$T_{ij,j}^{(w^l)} + \gamma^l w_i^l = 0 \qquad \text{in } \Omega^l,\ l = 0, ..., L, \tag{124a}$$

$$T_{ij}^{(w^0)}\mathcal{N}_j^0 = \frac{dg}{du_i} \quad \text{on } \bar{\Gamma}, \tag{124b}$$

$$w_i^l - w_i^0 = 0 \qquad \text{on } \Gamma^l,\ l = 1, ..., L, \tag{124c}$$

$$\left(T_{ij}^{(w^l)} - T_{ij}^{(w^0)}\right)\mathcal{N}_j^l = 0 \qquad \text{on } \Gamma^l,\ l = 1, ..., L, \tag{124d}$$

with

$$\frac{dg}{du_i} = \sum_{r=1}^{R} \left(u_i(x_r) - \ \hat{u}_i(x_r)\right)\delta(x - x_r). \tag{125}$$

This result is analogous to acoustics, in that the force term requires the forward solutions of the primary wave field u and its adjoint w, which is driven by the misfit between actual and simulated measurements. In the following section, it is further clarified how these minimizing equations are implemented as a numerical algorithm.

4.3.2 Numerical Algorithm

We now have a closer look at the obtained minimizing equations and how they lead to a practical inversion algorithm. In the remainder of this thesis, we consider only one unknown shape $\Gamma = \Gamma^1$ ($L = 1$), noting that the extension to the general case with several inclusions is straightforward because we have computed the shape derivatives for the multiple inclusion case. Eq. (114) and Eq. (121) are curve evolution equations driven by a force term F, which depends on the problem and the wave propagation model (e.g., acoustics or elastodynamics) under consideration.

This interface evolution is implemented using the level set method introduced in Section 1.3. Essentially, the algorithm updates the interface iteratively according to

$$\Gamma(k+1) = \Gamma(k) + \Delta t\, F(k)\mathcal{N}(k), \tag{126}$$

where Δt is the step size and k is the time iteration index. In practice, using the level set method, $\Gamma(k)$ is the zero level set of the level set function $\psi(k)$, i.e., $\Gamma(k) = \{x \in \Omega : \psi(x,k) = 0\}$, and $\psi(k)$ is evolved iteratively according

$$\psi(k+1) = \psi(k) - \Delta t\, F(k)\|\nabla\psi(k)\|. \tag{127}$$

In addition, a curve smoothness regularization term (see Section 1.3) weighted by α is added to increase robustness to local minima.

Given that level set interface evolution for a given force term F is a well-established procedure, the major remaining task is the computation of the force term $F(k)$ at each iteration step. In both acoustics and elastodynamics, F requires the displacement fields of the primary wave u and its adjoint w, which are solutions to boundary value problems (BVPs). Moreover, observe that w depends on u through $g(u)$, necessitating sequential computation of u and w. In order to solve these BVPs, several numerical solvers put themselves forward. We choose the Method of Fundamental Solutions (MFS) [83] because it is a *meshfree* numerical solver. Numerical solvers and in particular the MFS will be given particular attention in Section 4.4. The overall algorithm is sketched in Figure 46.

Level sets do not require meshing, so choosing a meshfree PDE solver makes the whole inversion algorithm completely meshfree. To our knowledge there do not exist any meshfree shape-based imaging methods in the literature. The absence of meshing simplifies the implementation of these algorithms especially in future three-dimensional applications.

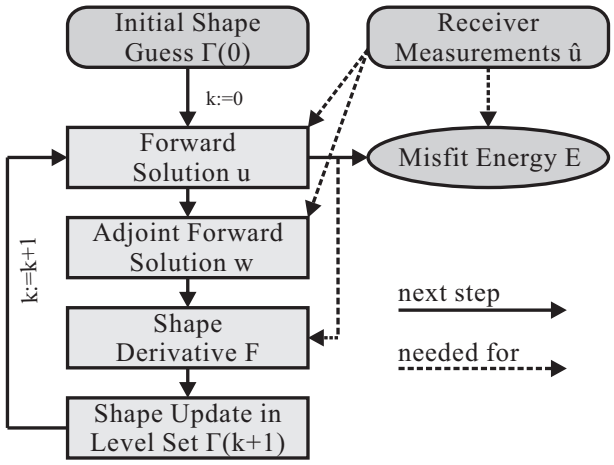


Figure 46: The adjoint active contour inversion algorithm. After guessing an initial shape $\Gamma(0)$, it is evolved iteratively to decrease the misfit energy E. In each iteration, acoustic solutions for the physical displacement field u and the adjoint displacement field w are simulated using the Method of Fundamental Solutions (MFS). These displacement fields are required for the computation of the force term F of the shape derivative. The force term drives the curve evolution, which is implemented in level set formulation.

4.4 Method of Fundamental Solutions

As discussed in Section 4.3, the adjoint method for inverse scattering requires solutions to acoustic and elastodynamic boundary value problems. Analytical solutions to these equations (and essentially any PDE) exist only for contrived cases with simple geometries for the boundaries such as circles. Since our approach uses active contour evolution to describe arbitrary shapes, numerical solutions are needed. As already mentioned, our method of choice is the Method of Fundamental Solutions (MFS) [82] because it is a method that does not require meshes nor grids, and can be easily combined with the data structure of active contours: It is straightforward to extract an (unordered) set of curve points with subpixel accuracy from an active contour in level set formulation by (e.g., linearly) interpolating the location of the zero level set. In two dimensions, and especially in three dimensions, it takes serious effort to extract (boundary or volume) elements from this unordered list of points, and will

require some sort of connection information between the points. The MFS, on the other hand, does not need any information about the connection between points.

We will formulate the MFS in detail for the Helmholtz equation in Section 4.4.2, and we then sketch the approach for the elastodynamic problem in Section 4.4.5. In the following section, however, other popular numerical solvers are reviewed briefly.

4.4.1 Review of Numerical Boundary Value Problem Solvers

Finite Difference Methods (FDM) are the most intuitive approach to approximating a PDE in discretized space. The PDE's derivatives are discretized and the resulting local conditions for each cell comprise a large system of equations. Numerous finite difference schemes have been proposed for solving the Helmholtz equation, e. g. [116] and most recently [104], where the PDE is approximated up to the sixth order. For linear PDEs, theoretical discretization error bounds can be obtained by Taylor expansion techniques. Even though this method is intuitive, it is fairly inefficient and especially for non-rectangular domains, incorporating boundary conditions is a very challenging task.

A similar approach are Finite Volume Methods (FVM), which are most extensively used in conservation processes such as fluid flow, where the concept of a control volume, in which a certain quantity like mass is preserved, plays a central role. Space is discretized in finite volumes and the volume average of the quantity is considered as opposed to point values in the FDM. This approach allows for a more insightful design of the scheme as it is modeled immediately after the physical conservation process. For example, for the Helmholtz equation one can use the fact that $\Delta u(x) = \nabla \cdot \nabla u(x)$ and employ the divergence theorem on the integral over the PDE to obtain an equation that can be interpreted as a conservation form:

$$\int_\Omega \Delta u + k^2 u \, dx = \int_\Gamma \nabla u \cdot \mathcal{N} \, ds + \int_\Omega k^2 u \, dx = 0. \tag{128}$$

In words, the average cell value is determined by the normal derivative of the quantity

along the boundary. Based on this conservation form, finite volume methods have been developed. A recent treatment—not only for elliptic equations—can be found in [42].

As mentioned above, FDM and FVM are not as popular as the methods discussed next, the main reason being that complicated geometries and boundary conditions are hard to implement in FDM and FVM. At the cost of more intricate mathematics, the Finite Element Method (FEM) and Boundary Element Method (BEM) yield equations that handle geometry and boundary conditions more easily using the element structure of those methods.

The FEM is the standard tool in structural mechanics and widely used in industrial applications. Recent research studies on FEM for the Helmholtz equation consider the problem of high frequency solutions, for which the standard FEM becomes very inaccurate [1]. A comprehensive introduction to FEM is given by [133], the authors being some of the early pioneers of the FEM. At the beginning of the FEM stands the discretization of the volume into a mesh of finite elements. The solution of the PDE on each element is approximated by polynomials, which, after invoking the variational form of the PDE and connectivity conditions between the elements, leads to a system of linear algebraic equations. The system matrix is sparse and has band-structure, since only neighbors of an element affect this element, and thus allows for a relatively efficient computation of the solution once the system matrix is created. The entries of the system matrix are integrals of the approximating function along the elements. FEM requires meshing of the whole volume, which is not always straightforward, and automatic mesh generation has a body of literature on its own [55]. This drawback was a major motivation for BEM to be developed, for which the meshed domain is one dimension lower and, therefore, less difficult to be meshed.

Two seminal publications [3] and [16] from 1977 mark the foundation of BEMs as we know them today. The core idea is to use integral representation formulas for

PDE solutions to transform the volumetric variational formulation of the FEM (integrals over the domain) into a surface formulation (integrals over the boundary of the domain). This approach requires knowledge of the fundamental solution or Green's function of the PDE and is therefore analytically more demanding than the FEM. As a reward, only the surface of the domain has to be discretized. Disadvantages of the BEM include the need for integration methods that can handle singular fundamental solution functions, as well as expensive numerical linear system solvers because the matrix is fully populated. Concerning the last point, one needs to keep in mind that only boundary points are discretized, meaning that the matrix is by an order smaller compared to FEM, where the whole volume is discretized. Details on the BEM can be found, for example, in [53].

Finally, a less known group of methods are collocation methods or Trefftz methods [89], which are meshfree solvers. As opposed to element methods, where nodes are connected via a structured mesh, boundaries in meshfree methods are represented by an unstructured and unsorted cluster of nodes. In what [82] refers to as indirect formulation, the solution is written as a weighted sum of basis functions that satisfy the PDE analytically. Then, the goal is to find the function in the span of those basis functions that match the boundary conditions best. As basis functions, any free space solution, in particular T-complete functions [66] or fundamental solutions, can be considered. Boundary conditions can be approximated using collocation, least squares residual or Galerkin formulation.

Obviously, BEM and collocation methods require some analytical niceness of the problem for analytical fundamental solutions to exist. But when this is the case, they can provide substantial benefit over the FEM as mentioned above. Collocation methods and BEMs are similar in many ways, and so collocation methods offer an alternative in most cases where BEM can be applied. In the following sections, the

Method of Fundamental Solutions (MFS), a classical collocation method, is formulated for our problems at hand.

4.4.2 Formulation for Acoustics

In this presentation of the MFS for acoustics, we limit ourselves to one inclusion, i.e., there is a domain Ω^1 inside a bounded domain Ω^0 as shown in Figure 45. Therefore, the goal is to numerically solve the PDE

$$-d_0\Delta u^0 + k_0^2 u^0 = -\sum_{s=1}^{S} \tilde{a}^s \delta(x - \tilde{\xi}^s), \quad x \in \Omega^0, \tag{129a}$$

$$-d_1\Delta u^1 + k_1^2 u^1 = 0, \qquad x \in \Omega^1, \tag{129b}$$

where we assume that the source term $S(x)$ is comprised of S point sources (modeled as Dirac impulses) located at $\{\tilde{\xi}^s\}_{s=1}^S \subset \Omega^0$. Throughout this derivation we denote $q^i = \partial u^i/\partial \mathcal{N}^i$, $i = 0, 1$ to be the flux, where $\mathcal{N}^i$ is the unit outward normal vector of Ω^i. Then, we require the boundary conditions

$$\lambda u^0(x) + (1-\lambda) q^0(x) = v_b(x), \qquad x \in \bar{\Gamma}, \tag{130a}$$

$$u^0(x) = u^1(x), \qquad x \in \Gamma^1, \tag{130b}$$

$$d_0 q^0(x) = -d_1 q^1(x), \quad x \in \Gamma^1, \tag{130c}$$

where $v_b(x)$ is given, as for example in the equations for the adjoint field Eq. (111b). Throughout, define

$$v^0(x) := \lambda u^0(x) + (1-\lambda) q^0(x) \tag{131}$$

for convenience and define the boundary residuals as

$$r_1(x) := v^0(x) - v_b(x), \qquad x \in \bar{\Gamma}, \tag{132a}$$

$$r_2(x) := u^0(x) - u^1(x), \qquad x \in \Gamma^1, \tag{132b}$$

$$r_3(x) := d_0 q^0(x) + d_1 q^1(x), \quad x \in \Gamma^1. \tag{132c}$$

The MFS deserves its name because it writes the numerical solution as a weighted sum of fundamental solutions. A fundamental solution $u^{j*}(x;\xi)$, $j = 0, 1$ located at ξ solves the PDE

$$-d_j \Delta u^{j*} + k_j^2 u^{j*} = -\delta(x - \xi) \tag{133}$$

in free space. The analytical fundamental solution for $d_j = -1$, i.e., the standard Helmholtz equation, is derived in Appendix C.1 and stated here as

$$u^{j*}(x;\xi) = \frac{i}{4} H_0^{(1)}(k_j \|x - \xi\|), \tag{134}$$

where i is the complex imaginary unit and $H_0^{(1)}$ is the Hankel function of the first kind and order 0. Back to the MFS, the approximate solutions u^0 and u^1 are written as linear combinations of fundamental solutions, also called poles, as

$$u^0(x) = \sum_{n=1}^{P_0} a^{0n} u^{0*}(x;\xi^{0n}) + \sum_{s=1}^{S} \tilde{a}^s u^{0*}(x;\tilde{\xi}^s), \tag{135a}$$

$$u^1(x) = \sum_{n=1}^{P_1} a^{1n} u^{1*}(x;\xi^{1n}), \tag{135b}$$

where $\xi^{in} \notin \Omega^i$. In this way, the PDE is satisfied analytically within Ω^i. P_i is the number of poles spanning the solution of Ω_i, and $P = P_0 + P_1$ is the total number of poles. The pole locations are a design parameter, and this issue will be discussed further in Section 4.4.3. Given the pole location, the goal is to find the amplitudes a^{0n}, a^{1n} to satisfy the boundary conditions. Let us vectorize the equations to improve readability. Calling

$$\boldsymbol{a}^0 := \left(a^{01},\, a^{02},\, a^{03}, \cdots,\, a^{0P_0}\right)^T, \tag{136a}$$

$$\boldsymbol{a}^1 := \left(a^{11},\, a^{12},\, a^{13}, \cdots,\, a^{1P_1}\right)^T, \tag{136b}$$

$$\boldsymbol{u}^{0*}(x) := \left(u^{0*}(x;\xi^{01}),\, u^{0*}(x;\xi^{02}),\, u^{0*}(x;\xi^{03}), \cdots,\, u^{0*}(x;\xi^{0P_0})\right)^T, \tag{136c}$$

$$\boldsymbol{u}^{1*}(x) := \left(u^{1*}(x;\xi^{11}),\, u^{1*}(x;\xi^{12}),\, u^{1*}(x;\xi^{13}), \cdots,\, u^{1*}(x;\xi^{1P_1})\right)^T, \tag{136d}$$

$$\boldsymbol{q}^{0*}(x) := \left(q^{0*}(x;\xi^{01}),\, q^{0*}(x;\xi^{02}),\, q^{0*}(x;\xi^{03}), \cdots,\, q^{0*}(x;\xi^{0P_0})\right)^T, \tag{136e}$$

$$\boldsymbol{q}^{1*}(x) := \left(q^{1*}(x;\xi^{11}),\, q^{1*}(x;\xi^{12}),\, q^{1*}(x;\xi^{13}), \cdots,\, q^{1*}(x;\xi^{1P_1})\right)^T, \tag{136f}$$

and

$$u_s(x) := \sum_{s=1}^{S} \tilde{a}^s u^{0*}(x; \tilde{\xi}^s), \tag{137a}$$

$$q_s(x) := \sum_{s=1}^{S} \tilde{a}^s q^{0*}(x; \tilde{\xi}^s), \tag{137b}$$

$$v_s(x) := \lambda u_s(x) + (1-\lambda) q_s(x), \tag{137c}$$

$$v_{sb}(x) := v_s(x) - v_b(x) \tag{137d}$$

we have

$$v^0(x) = \boldsymbol{v}^{0*}(x)^T \boldsymbol{a}^0 + v_s(x), \tag{138a}$$

$$u^0(x) = \boldsymbol{u}^{0*}(x)^T \boldsymbol{a}^0 + u_s(x), \quad u^1(x) = \boldsymbol{u}^{1*}(x)^T \boldsymbol{a}^1, \tag{138b}$$

$$q^0(x) = \boldsymbol{q}^{0*}(x)^T \boldsymbol{a}^0 + q_s(x), \quad q^1(x) = \boldsymbol{q}^{1*}(x)^T \boldsymbol{a}^1, \tag{138c}$$

and

$$r_1(x) = \boldsymbol{v}^{0*}(x)^T \boldsymbol{a}^0 + v_{sb}(x), \tag{139a}$$

$$r_2(x) = \boldsymbol{u}^{0*}(x)^T \boldsymbol{a}^0 - \boldsymbol{u}^{1*}(x)^T \boldsymbol{a}^1 + u_s(x), \tag{139b}$$

$$r_3(x) = d_0 \boldsymbol{q}^{0*}(x)^T \boldsymbol{a}^0 + d_1 \boldsymbol{q}^{1*}(x)^T \boldsymbol{a}^1 + d_0 q_s(x). \tag{139c}$$

The goal now is to minimize the residuals $r_1(x)$, $r_2(x)$, and $r_3(x)$ on a discrete set of collocation points on the respective boundaries. A quite general approach is the weighted least squares method, from which several special cases will be derived.

Let there be N_0 collocation points x^{0i}, $i = 1, .., N_0$ on $\bar{\Gamma}$ and N_1 collocation points x^{1i}, $i = 1, .., N_1$ on Γ^1, which we collect in the vectors

$$\boldsymbol{x}^0 := \left(x^{01}, x^{02}, x^{03}, \cdots, x^{0N_0}\right)^T, \tag{140a}$$

$$\boldsymbol{x}^1 := \left(x^{11}, x^{12}, x^{13}, \cdots, x^{1N_1}\right)^T. \tag{140b}$$

How these collocation points are obtained in practice is discussed in Section 4.4.3. Our goal is to minimize the weighted squared residuals evaluated at the collocation

points

$$\min_{\boldsymbol{a}^0,\boldsymbol{a}^1} \alpha_1 \sum_{n=1}^{N_0} w_1^n \left(r_1(x^{0n})\right)^2 + \alpha_2 \sum_{n=1}^{N_1} w_2^n \left(r_2(x^{1n})\right)^2 + \alpha_3 \sum_{n=1}^{N_1} w_3^n \left(r_3(x^{1n})\right)^2, \tag{141}$$

for given residual weights α_1, α_2, $\alpha_3 \geq 0$ and collocation weights

$$\boldsymbol{w}_1^T := \left(w_1^1,\, w_1^2,\, w_1^3, \cdots,\, w_1^{N_0}\right) \geq 0, \tag{142a}$$

$$\boldsymbol{w}_2^T := \left(w_2^1,\, w_2^2,\, w_2^3, \cdots,\, w_2^{N_1}\right) \geq 0, \tag{142b}$$

$$\boldsymbol{w}_3^T := \left(w_3^1,\, w_3^2,\, w_3^3, \cdots,\, w_3^{N_1}\right) \geq 0. \tag{142c}$$

We define $N = N_0 + 2N_1$, the number of terms in Eq. (141). Again, we vectorize to obtain

$$\min_{\boldsymbol{a}_0,\boldsymbol{a}_1} \alpha_1 \boldsymbol{r}_1^T \boldsymbol{W}_1 \boldsymbol{r}_1 + \alpha_2 \boldsymbol{r}_2^T \boldsymbol{W}_2 \boldsymbol{r}_2 + \alpha_3 \boldsymbol{r}_3^T \boldsymbol{W}_3 \boldsymbol{r}_3, \tag{143}$$

where

$$\boldsymbol{r}_1 := r_1(\boldsymbol{x}^0), \quad \boldsymbol{r}_2 := r_2(\boldsymbol{x}^1), \quad \boldsymbol{r}_3 := r_3(\boldsymbol{x}^1), \tag{144}$$

and

$$\boldsymbol{W}_1 := \mathrm{diag}(\boldsymbol{w}_1), \quad \boldsymbol{W}_2 := \mathrm{diag}(\boldsymbol{w}_2), \quad \boldsymbol{W}_3 := \mathrm{diag}(\boldsymbol{w}_3), \tag{145}$$

where $\mathrm{diag}(\boldsymbol{v})$ denotes the diagonal square matrix that has the entries of the vector $\boldsymbol{v}$ on its diagonal. We now express the objective function in terms of the amplitudes $\boldsymbol{a}^0$ and $\boldsymbol{a}^1$ and obtain

$$\boldsymbol{r}_1 = v^0(\boldsymbol{x}^0) = \boldsymbol{v}^{0*}(\boldsymbol{x}^0)^T \boldsymbol{a}^0 + v_{sb}(\boldsymbol{x}^0), \tag{146a}$$

$$\boldsymbol{r}_2 = u^0(\boldsymbol{x}^1) - u^1(\boldsymbol{x}^1) = \boldsymbol{u}^{0*}(\boldsymbol{x}^1)^T \boldsymbol{a}^0 + u_s(\boldsymbol{x}^1) - \boldsymbol{u}^{1*}(\boldsymbol{x}^1)^T \boldsymbol{a}^1, \tag{146b}$$

$$\boldsymbol{r}_3 = d_0 q^0(\boldsymbol{x}^1) + d_1 q^1(\boldsymbol{x}^1) = d_0 \boldsymbol{q}^{0*}(\boldsymbol{x}^1)^T \boldsymbol{a}^0 + d_0 q_s(\boldsymbol{x}^1) + d_1 \boldsymbol{q}^{1*}(\boldsymbol{x}^1)^T \boldsymbol{a}^1. \tag{146c}$$

Here, the evaluation of the fundamental solution vectors indicated by * at the collocation point vectors produces matrices, which are to be interpreted, for example, as

$$[\boldsymbol{u}^{0*}(\boldsymbol{x}^1)^T]_{(i,j)} = u^{0*}(x^{1i}; \xi^{0j}) \tag{147}$$

and analogous for the other terms. This means that the collocation point does not change within a row, whereas the pole does not change within a column. We abbreviate these matrices as follows

$$\boldsymbol{V}_0^* := \boldsymbol{v}^{0*}(\boldsymbol{x}^0) \in \mathbb{R}^{P_0 \times N_0}, \tag{148a}$$

$$\boldsymbol{U}_0^* := \boldsymbol{u}^{0*}(\boldsymbol{x}^1) \in \mathbb{R}^{P_0 \times N_1}, \quad \boldsymbol{U}_1^* := \boldsymbol{u}^{1*}(\boldsymbol{x}^1) \in \mathbb{R}^{P_1 \times N_1}, \tag{148b}$$

$$\boldsymbol{Q}_0^* := \boldsymbol{q}^{0*}(\boldsymbol{x}^1) \in \mathbb{R}^{P_0 \times N_1}, \quad \boldsymbol{Q}_1^* := \boldsymbol{q}^{1*}(\boldsymbol{x}^1) \in \mathbb{R}^{P_1 \times N_1} \tag{148c}$$

as well as the vectors

$$\boldsymbol{v}_{sb} := v_{sb}(\boldsymbol{x}^0), \quad \boldsymbol{u}_s := u_s(\boldsymbol{x}^1), \quad \boldsymbol{q}_s := q_s(\boldsymbol{x}^1). \tag{149}$$

So, the residual vectors are given in this matrix notation as

$$\boldsymbol{r}_1 = \boldsymbol{V}_0^{*T}\boldsymbol{a}^0 + \boldsymbol{v}_{sb}, \tag{150a}$$

$$\boldsymbol{r}_2 = \boldsymbol{U}_0^{*T}\boldsymbol{a}^0 - \boldsymbol{U}_1^{*T}\boldsymbol{a}^1 + \boldsymbol{u}_s, \tag{150b}$$

$$\boldsymbol{r}_3 = d_0\boldsymbol{Q}_0^{*T}\boldsymbol{a}^0 + d_1\boldsymbol{Q}_1^{*T}\boldsymbol{a}^1 + d_0\boldsymbol{q}_s. \tag{150c}$$

Now substituting into Eq. (143), taking derivatives with respect to $\boldsymbol{a}^0$ and $\boldsymbol{a}^1$, and setting the derivatives to zero, yields

$$\bar{\boldsymbol{A}}\boldsymbol{a} = \bar{\boldsymbol{b}}, \tag{151}$$

where

$$\bar{\boldsymbol{A}} = \begin{bmatrix} \alpha_1\boldsymbol{V}_0^*\boldsymbol{W}_1\boldsymbol{V}_0^{*T} + \alpha_2\boldsymbol{U}_0^*\boldsymbol{W}_2\boldsymbol{U}_0^{*T} + \alpha_3 d_0^2\boldsymbol{Q}_0^*\boldsymbol{W}_3\boldsymbol{Q}_0^{*T} & -\alpha_2\boldsymbol{U}_0^*\boldsymbol{W}_2\boldsymbol{U}_1^{*T} + \alpha_3 d_0 d_1\boldsymbol{Q}_0^*\boldsymbol{W}_3\boldsymbol{Q}_1^{*T} \\ -\alpha_2\boldsymbol{U}_1^*\boldsymbol{W}_2\boldsymbol{U}_0^{*T} + \alpha_3 d_0 d_1\boldsymbol{Q}_1^*\boldsymbol{W}_3\boldsymbol{Q}_0^{*T} & \alpha_2\boldsymbol{U}_1^*\boldsymbol{W}_2\boldsymbol{U}_1^{*T} + \alpha_3 d_1^2\boldsymbol{Q}_1^*\boldsymbol{W}_3\boldsymbol{Q}_1^{*T} \end{bmatrix}, \tag{152a}$$

$$\boldsymbol{a} = \begin{bmatrix} \boldsymbol{a}^0 \\ \boldsymbol{a}^1 \end{bmatrix}, \quad \bar{\boldsymbol{b}} = \begin{bmatrix} -\alpha_1\boldsymbol{V}_0^*\boldsymbol{W}_1\boldsymbol{v}_{sb} - \alpha_2\boldsymbol{U}_0^*\boldsymbol{W}_2\boldsymbol{u}_s - \alpha_3 d_0^2\boldsymbol{Q}_0^*\boldsymbol{W}_3\boldsymbol{q}_s \\ \alpha_2\boldsymbol{U}_1^*\boldsymbol{W}_2\boldsymbol{u}_s - \alpha_3 d_0 d_1\boldsymbol{Q}_1^*\boldsymbol{W}_3\boldsymbol{q}_s \end{bmatrix}. \tag{152b}$$

The above derivation shows that solving the linear system Eq. (151) returns the minimizer of Eq. (141). Furthermore It is interesting to observe that if $N \geq P$, i.e., if there are not more poles than collocation residuals, solving Eq. (151) is equivalent

to solving the following linear system in the least square sense

$$\boldsymbol{A}\boldsymbol{a} := \begin{bmatrix} \sqrt{\alpha_1}\sqrt{\boldsymbol{W}_1}\boldsymbol{V}_0^{*T} & \boldsymbol{0} \\ \sqrt{\alpha_2}\sqrt{\boldsymbol{W}_2}\boldsymbol{U}_0^{*T} & -\sqrt{\alpha_2}\sqrt{\boldsymbol{W}_2}\boldsymbol{U}_1^{*T} \\ \sqrt{\alpha_3}d_0\sqrt{\boldsymbol{W}_3}\boldsymbol{Q}_0^{*T} & \sqrt{\alpha_3}d_1\sqrt{\boldsymbol{W}_3}\boldsymbol{Q}_1^{*T} \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_0 \\ \boldsymbol{a}_1 \end{bmatrix} = \begin{bmatrix} -\sqrt{\alpha_1}\sqrt{\boldsymbol{W}_1}\boldsymbol{v}_{sb} \\ -\sqrt{\alpha_2}\sqrt{\boldsymbol{W}_2}\boldsymbol{u}_s \\ -\sqrt{\alpha_3}d_0\sqrt{\boldsymbol{W}_3}\boldsymbol{q}_s \end{bmatrix} =: \boldsymbol{b}, \tag{153}$$

where $\boldsymbol{A} \in \mathbb{C}^{N\times P}$. In other words, it is easy to check that

$$\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{a} = \boldsymbol{A}^T\boldsymbol{b} \tag{154}$$

yields the same linear system as Eq. (151), i.e., $\bar{\boldsymbol{A}} = \boldsymbol{A}^T\boldsymbol{A}$ and $\bar{\boldsymbol{b}} = \boldsymbol{A}^T\boldsymbol{b}$. This last system of equations simply sets the boundary conditions at collocation points to zero. If there is an exact solution, the residuals will be zero, that is the numerical solution exactly satisfies the boundary conditions at the collocation points. In case the system is over-determined, i.e., if $N > P$, we seek the least square solution, which turns out to be the same as minimizing the squared residuals in Eq. (141). If the system happens to be under-determined, e.g., when $N = P$ and two poles happen to be extremely close to each other, we seek the exact solution that has minimum norm. Next, we consider some special cases of this general weighted collocation method.

Exact Collocation Method. The exact collocation method requires $N = P$, i.e., $\boldsymbol{A}$ is square, and $\boldsymbol{A}$ must be non-singular. Then, Eq. (153) is solved directly without using least squares. In this case, there is a unique solution and we can divide each row by the appropriate weighting constants α and w to obtain

$$\boldsymbol{A}\boldsymbol{a} := \begin{bmatrix} \boldsymbol{V}_0^{*T} & \boldsymbol{0} \\ \boldsymbol{U}_0^{*T} & -\boldsymbol{U}_1^{*T} \\ \boldsymbol{Q}_0^{*T} & \boldsymbol{Q}_1^{*T} \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_0 \\ \boldsymbol{a}_1 \end{bmatrix} = \begin{bmatrix} -\boldsymbol{v}_{sb} \\ -\boldsymbol{u}_s \\ -\boldsymbol{q}_s \end{bmatrix} =: \boldsymbol{b} \tag{155}$$

(Unweighted) Least Squares Collocation Method. If $N > P$ and $\boldsymbol{W}_i = \boldsymbol{I}$, $i = 1, 2, 3$, where $\boldsymbol{I}$ is the identity matrix of appropriate size, we refer to Eq. (151) as

Least Squares Collocation Method. While the weights α_1, α_2, and α_3 still have to be chosen in order to balance the different boundary conditions in the least squares sum, and to account for the unit discrepancy between the quantity u and its flux q, setting all the w-weights to 1 makes no distinction between the collocation points.

Least Squares Integral Method. We now show a connection between the weighted least squares collocation method and a similar method that aims at minimizing the *continuous* squared residual along the boundaries. Consider the integral of the squared residuals along the boundaries

$$\min_{\boldsymbol{a}_0,\boldsymbol{a}_1} \alpha_1 \int_{\bar{\Gamma}} r_1(x)^2 \, d\Gamma + \alpha_2 \int_{\Gamma^1} r_2(x)^2 \, d\Gamma + \alpha_3 \int_{\Gamma^1} r_3(x)^2 \, d\Gamma, \tag{156}$$

which is to be minimized. Using Eq. (139) and minimizing the above equation, one obtains

$$\bar{\boldsymbol{A}}\boldsymbol{a} = \bar{\boldsymbol{b}}, \tag{157}$$

where

$$\bar{\boldsymbol{A}} = \begin{bmatrix} \bar{\boldsymbol{A}}_{(1,1)} & \bar{\boldsymbol{A}}_{(1,2)} \\ \bar{\boldsymbol{A}}_{(2,1)} & \bar{\boldsymbol{A}}_{(2,2)} \end{bmatrix}, \tag{158}$$

with

$$\bar{\boldsymbol{A}}_{(1,1)} = \alpha_1 \int_{\bar{\Gamma}} \boldsymbol{v}^{0*} \boldsymbol{v}^{0*^T} \, d\Gamma + \alpha_2 \int_{\Gamma^1} \boldsymbol{u}^{0*} \boldsymbol{u}^{0*^T} \, d\Gamma + \alpha_3 d_0^2 \int_{\Gamma^1} \boldsymbol{q}^{0*} \boldsymbol{q}^{0*^T} \, d\Gamma, \tag{159a}$$

$$\bar{\boldsymbol{A}}_{(1,2)} = -\alpha_2 \int_{\Gamma^1} \boldsymbol{u}^{0*} \boldsymbol{u}^{1*^T} \, d\Gamma + \alpha_3 d_0 d_1 \int_{\Gamma^1} \boldsymbol{q}^{0*} \boldsymbol{q}^{1*^T} \, d\Gamma, \tag{159b}$$

$$\bar{\boldsymbol{A}}_{(2,1)} = -\alpha_2 \int_{\Gamma_1} \boldsymbol{u}_1^{*} \boldsymbol{u}_0^{*^T} \, d\Gamma + \alpha_3 d_0 d_1 \int_{\Gamma_1} \boldsymbol{q}_1^{*} \boldsymbol{q}_0^{*^T} \, d\Gamma, \tag{159c}$$

$$\bar{\boldsymbol{A}}_{(2,2)} = \alpha_2 \int_{\Gamma_1} \boldsymbol{u}_1^{*} \boldsymbol{u}_1^{*^T} \, d\Gamma + \alpha_3 d_1^2 \int_{\Gamma_1} \boldsymbol{q}_1^{*} \boldsymbol{q}_1^{*^T} \, d\Gamma, \tag{159d}$$

and

$$\bar{\boldsymbol{b}} = \begin{bmatrix} -\alpha_1 \int_{\bar{\Gamma}} \boldsymbol{v}^{0*} v_{sb} \, d\Gamma - \alpha_2 \int_{\Gamma^1} \boldsymbol{u}^{0*} u_s \, d\Gamma - \alpha_3 d_0^2 \int_{\Gamma^1} \boldsymbol{q}^{0*} q_s \, d\Gamma \\ \alpha_2 \int_{\Gamma^1} \boldsymbol{u}^{1*} u_s \, d\Gamma - \alpha_3 d_0 d_1 \int_{\Gamma^1} \boldsymbol{q}^{1*} q_s \, d\Gamma \end{bmatrix}. \tag{160}$$

Next, we would like to approximate the integrals along boundaries, by summing over

collocation points. That is, for example,

$$\int_{\bar{\Gamma}} \boldsymbol{v}^{0*}\boldsymbol{v}^{0*T}\, d\Gamma \approx \sum_{i=1}^{N_0} \boldsymbol{v}^{0*}(x^{0i})\boldsymbol{v}^{0*T}(x^{0i})\Delta\Gamma_i. \tag{161}$$

In the case of a boundary curve in two dimensions, where the collocation points are sorted by their connection, the integral measure $\Delta\Gamma_i$ for collocation point x^{0i} could be approximated by $(\|x^{0(i+1)} - x^{0i}\| + \|x^{0i} - x^{0(i-1)}\|)/2$. For level set representations of an interface, there are efficient ways to extract this integral measure from the level set function. In any case, letting

$$\boldsymbol{\delta}^0 := (\Delta\Gamma_1, \Delta\Gamma_2 \cdots,\ \Delta\Gamma_{N_0})^T \tag{162}$$

and $\boldsymbol{\Delta}_0 := \mathrm{diag}(\boldsymbol{\delta}^0)$, we obtain

$$\int_{\Gamma} \boldsymbol{v}^{0*}\boldsymbol{v}^{0*T}\, d\Gamma \approx \boldsymbol{V}_0^*\boldsymbol{\Delta}_0\boldsymbol{V}_0^{*T}, \tag{163}$$

where we use notation as in Eq. (148). Similarly, we obtain matrix expressions for the other integral terms and we find that

$$\bar{\boldsymbol{A}} \approx \begin{bmatrix} \alpha_1\boldsymbol{V}_0^*\boldsymbol{\Delta}_0\boldsymbol{V}_0^{*T} + \alpha_2\boldsymbol{U}_0^*\boldsymbol{\Delta}_1\boldsymbol{U}_0^{*T} + \alpha_3 d_0^2\boldsymbol{Q}_0^*\boldsymbol{\Delta}_1\boldsymbol{Q}_0^{*T} & -\alpha_2\boldsymbol{U}_0^*\boldsymbol{\Delta}_1\boldsymbol{U}_1^{*T} + \alpha_3 d_0 d_1\boldsymbol{Q}_0^*\boldsymbol{\Delta}_1\boldsymbol{Q}_1^{*T} \\ -\alpha_2\boldsymbol{U}_1^*\boldsymbol{\Delta}_1\boldsymbol{U}_0^{*T} + \alpha_3 d_0 d_1\boldsymbol{Q}_1^*\boldsymbol{\Delta}_1\boldsymbol{Q}_0^{*T} & \alpha_2\boldsymbol{U}_1^*\boldsymbol{\Delta}_1\boldsymbol{U}_1^{*T} + \alpha_3 d_1^2\boldsymbol{Q}_1^*\boldsymbol{\Delta}_1\boldsymbol{Q}_1^{*T} \end{bmatrix}, \tag{164a}$$

$$\bar{\boldsymbol{b}} \approx \begin{bmatrix} -\alpha_1\boldsymbol{V}_0^*\boldsymbol{\Delta}_0\boldsymbol{v}_{sb} - \alpha_2\boldsymbol{U}_0^*\boldsymbol{\Delta}_1\boldsymbol{u}_s - \alpha_3 d_0^2\boldsymbol{Q}_0^*\boldsymbol{\Delta}_1\boldsymbol{q}_s \\ \alpha_2\boldsymbol{U}_1^*\boldsymbol{\Delta}_1\boldsymbol{u}_s - \alpha_3 d_0 d_1\boldsymbol{Q}_1^*\boldsymbol{\Delta}_1\boldsymbol{q}_s \end{bmatrix}, \tag{164b}$$

which we identify with Eq. (151) when $\boldsymbol{W}_1 = \boldsymbol{\Delta}_0$ and $\boldsymbol{W}_2 = \boldsymbol{W}_3 = \boldsymbol{\Delta}_1$. So, a discrete approximation of the integral residual method is given by choosing the weights to be approximated integral measures of the corresponding collocation points.

4.4.3 Practical Considerations

Two design questions were left open in the MFS derivation above: The number and location of poles and the number and location of the collocation points. We start with the collocation points. Obviously, these points should lie on the interface that they represent. The outer boundary $\bar{\Gamma}$ does not move and can, therefore, be treated with special care to achieve even distances between the collocation points because

this is a favorable property for accuracy as is shown in Section 4.4.4. If this boundary is a curve in two-dimensions, these points can be obtained by uniformly sampling points on the curve. In three dimensions, this task becomes more cumbersome and very much dependent on the data structure of the boundary. For the moving active contour interface, we pursue a different approach. One can easily (in any dimension) interpolate the zero level set crossings with the computational grid lines, which provide sub-pixel accurate locations of interface points. The number of thus extracted points can be increased or decreased by choosing a finer or coarser computational grid. Note that with this method, however, the extracted points are not equidistant or evenly distributed. This less favorable property is the price paid for the efficient and easy extraction of collocation points from the level set. There is a bound for the maximum distance to the nearest point, however, which is the distance between two opposite vertices in a grid cell. It is also clear that the number of collocation points is dynamic, i.e., changing with the curve evolution, whereas the number of collocation points on the outer boundary is usually fixed and determined before the inversion algorithms starts.

Generally, a pole is assigned to a collocation point. Note that this does not imply that each collocation point has an assigned pole necessarily. For exact collocation, for example, the following applies: On $\bar{\Gamma}$, each collocation point is associated with one pole for Ω^0 (centered outside Ω), and on Γ^1, each collocation point is associated with two poles, one for Ω^0 (centered within Ω^1) and one for Ω^1 (centered within Ω^0). Hence, $P_0 = N_0 + N_1$ and $P_1 = N_1$. A pole is located along the line perpendicular to the curve and passing through the associated collocation point, i.e., along the normal direction of the surface. The normal direction at collocation points is easily extracted from level sets by computing their numerical gradient at these location. Figure 47 illustrates this setup. In some cases, we may choose to not associate some collocation points with a pole, e.g., one could have $P \approx N/K$ where $K > 1$, to make the system

under-determined and, thus, more robust. A higher value for K is usually achieved by increasing the number of collocation points (while keeping the number of poles fixed) to increase accuracy.

Thus, there are three remaining design parameters: The number of collocation points (controlled through the grid size), the number of poles (controlled through the constant K), and the the pole distance to the surface, which may, in fact, vary for different interfaces, e.g., the distance for the outer boundary $\bar{\Gamma}$ may be chosen larger than for the inner interface Γ^l. The pole distances are usually chosen to make the condition number of the system matrix Eq. (153) attain a moderate value. If the condition number is too high, the results might become sensitive to noise, even though it has been reported that the numerical solutions can still be surprisingly accurate for high condition numbers [43]. These issues will be further analyzed in the next section, where the MFS is tested on a simple example where an analytical solution

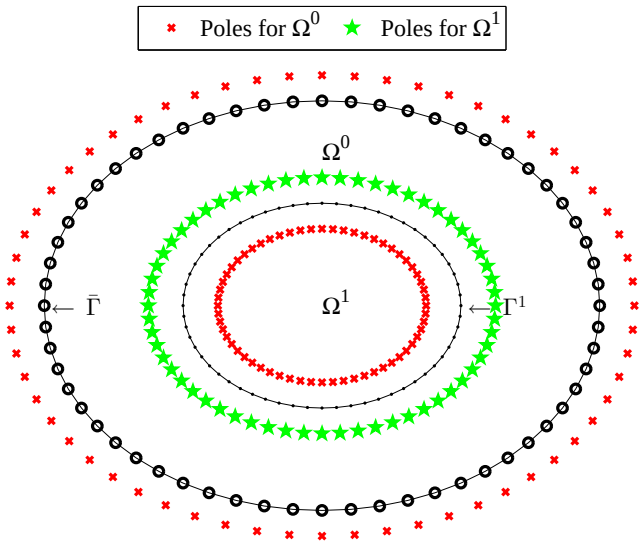


Figure 47: Sample setup for collocation points and poles in the MFS. Collocation points are sampled from the respective interface, whereas poles are placed outside the respective domain at a certain distance from the interface.

exists. For the exact collocation method, we investigate the influence of the number of collocation points, the pole distance and the effect of uneven collocation points on the accuracy of the numerical solution.

4.4.4 Experimental Analysis for Acoustics

In this section, the MFS with exact collocation is implemented and evaluated for a simple geometry where an analytical solution exists. Influence on numerical accuracy is investigated for changes in

1. number of collocation points,
2. pole distance,
3. non-uniform spacing between collocation nodes.

Consider a setup where both $\bar{\Gamma}$ and Γ^1 are circular with radii $R^0 = 4$, $R^1 = 2$ and centered at the origin. The test BVP is given by

$$\Delta u^i(x) + k_i^2 u^i(x) = 0, \qquad x \in \Omega^i, \quad i = 0, 1 \tag{165a}$$

$$\lambda u^0(x) + (1-\lambda)\frac{\partial u^0(x)}{\partial \mathcal{N}^0} = \cos(n\theta), \quad x \in \bar{\Gamma} \tag{165b}$$

$$u^1(x) - u^0(x) = 0, \qquad x \in \Gamma^1 \tag{165c}$$

$$\frac{\partial u^1(x)}{\partial \mathcal{N}^1} - \frac{\partial u^0(x)}{\partial \mathcal{N}^1} = 0, \qquad x \in \Gamma^1 \tag{165d}$$

for $k_0 = \sqrt{0.1}$, $k_1 = \sqrt{0.01}$, $\lambda = 0.5$, $n = 5$, and θ is the angle parameter for the circular boundary $\bar{\Gamma}$, i. e., $\theta(x) = \tan^{-1}(x_2/x_1)$ where $x = (x_1, x_2) \in \bar{\Gamma}$. First, the analytical solution for this problem is presented for future comparison to the numerical solution using the MFS.

4.4.4.1 Analytical Solution

In polar coordinates (r, θ), the homogeneous Helmholtz equation becomes

$$u_{,rr} + \frac{1}{r}u_{,r} + \frac{1}{r^2}u_{,\theta\theta} + k^2 u = 0. \tag{166}$$

Separation of variables leads to a solution of the form

$$u(r,\theta) = R(r)\Theta(\theta), \tag{167}$$

where $R(r)$ and $\Theta(\theta)$ are determined by the ordinary differential equations (ODE)

$$r^2 R^{''} + rR^{'} + (r^2k^2 - m^2)R = 0, \tag{168}$$

and

$$\Theta^{''} + m^2\Theta = 0, \tag{169}$$

for $m = 0, 1, 2, ...$ The general solutions of those ODEs are Bessel functions and sinusoidal functions, respectively, so that the general solution for Eq. (166) is written as

$$u(r,\theta) = \sum_{m=0}^{\infty} \left(a_m J_m(kr) + b_m Y_m(kr)\right) \left(c_m \sin(m\theta) + d_m \cos(m\theta)\right), \tag{170}$$

where J_m and Y_m are Bessel functions of the first and second kind, respectively, both of order m, and the constants a_m, b_m , c_m, and d_m have to be determined from the boundary conditions. Observe that $\partial u(r,\theta)/\partial\mathcal{N} = ru_{,r}(r,\theta)$, where the derivative $u_{,r}(r,\theta)$ is obtained from Eq. (170) as

$$u_{,r}(r,\theta) = \sum_{m=0}^{\infty} \left(a_m k J_m^{'}(kr) + b_m k Y_m^{'}(kr)\right) \left(c_m \sin(m\theta) + d_m \cos(m\theta)\right). \tag{171}$$

For Bessel functions, the following recursive formulas apply for the derivatives

$$J_p^{'}(kr) = \frac{1}{2}(J_{p-1}(kr) - J_{p+1}(kr)), \quad Y_p^{'}(kr) = \frac{1}{2}(Y_{p-1}(kr) - Y_{p+1}(kr)). \tag{172}$$

Thus, from boundary condition Eq. (165b), only terms involving $\cos(n\theta)$ must be retained and so

$$u^0(r,\theta) = \left(\tilde{a}_n J_n(k^0 r) + \tilde{b}_n Y_n(k^0 r)\right) \cos(n\theta). \tag{173}$$

Moreover, since Y_n is singular at the origin, this term must be eliminated in the inner domain in order to ensure a continuous solution. Hence,

$$u^1(r,\theta) = \tilde{c}_n J_n(k^1 r) \cos(n\theta). \tag{174}$$

We are left with finding the amplitudes $(\tilde{a}_n, \tilde{b}_n, \tilde{c}_n)$ from the boundary conditions in Eq. (165). The resulting system of equations is

$$\begin{bmatrix} \lambda J_n(k^0R^0)+(1-\lambda)J_n'(k^0R^0) & \lambda Y_n(k^0R^0)+(1-\lambda)Y_n'(k^0R^0) & 0 \\ J_n(k^0R^1) & Y_n(k^0R^1) & -J_n(k^1R^1) \\ J_n'(k^0R^1) & Y_n'(k^0R^1) & -J_n'(k^1R^1) \end{bmatrix} \begin{bmatrix} \tilde{a}_n \\ \tilde{b}_n \\ \tilde{c}_n \end{bmatrix} = 0, \tag{175}$$

which is readily solved to yield $(\tilde{a}_5, \tilde{b}_5, \tilde{c}_5) \approx (1.14 \cdot 10^3,\ 3.62 \cdot 10^{-8},\ 3.55 \cdot 10^5)$. Thus, the final solution is

$$u^0(r,\theta) = \left(1.14 \cdot 10^3 \cdot J_5(\sqrt{0.1}r) + 3.62 \cdot 10^{-8} \cdot Y_5(\sqrt{0.1}r)\right)\cos(5\theta), \tag{176a}$$

$$u^1(r,\theta) = 3.55 \cdot 10^5 \cdot J_5(\sqrt{0.01}r)\cos(5\theta). \tag{176b}$$

Note that the double precision solution of Eq. (175) for the coefficients $(\tilde{a}_n, \tilde{b}_n, \tilde{c}_n)$ is used when comparing numerical solutions to this true solution. The analytical solution is plotted in Figure 48 for illustration.

4.4.4.2 Methodology

Three test cases are considered in this section to validate the accuracy of the numerical solution and investigate the influence of parameter changes. Foremost, an increased number of collocation points or poles is expected to improve the numerical result,

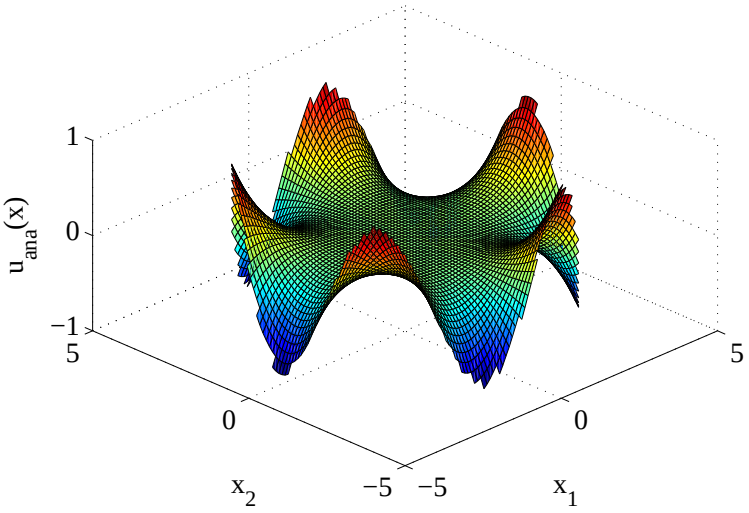


Figure 48: Analytical solution for the BVP Eq. (165).

which is confirmed in the first experiment. The second experiment elaborates on the first test by investigating the effect of the pole distances, which is shown to be a crucial parameter. Third, the distances between the collocation points are varied to be non-uniform, letting collocation points move close to each other. This is an interesting experiment from the application point of view, since the nodes extracted from the level set function are far from uniformly distributed along the curve.

Accuracy of the numerical solution $u(x)$ is evaluated through the maximum absolute error compared to the analytical solution $u_{\text{ana}}(x)$,

$$E_{\max} = \max_{x \in \Omega^0 \cup \Omega^1} |u(x) - u_{\text{ana}}(x)|. \tag{177}$$

Also of interest is the condition number (the ratio of the largest to the smallest singular value) of the MFS system matrix $\boldsymbol{A}$ in Eq. (153) since one would expect inaccurate solutions for high condition numbers. It is shown, however, that interestingly even for high condition numbers the solution can be accurate, as was observed previously by [43], [83], [84].

4.4.4.3 Experiment 1: Number of Collocation Points

The first experiment analyzes the influence of the number of the collocation points. The outer and the inner boundary are discretized with $\bar{N}$ collocation points each and a pole distance $d_p = 0.5$ is chosen. Figure 49 shows the resulting errors and condition numbers. As expected, the relationship between error and $\bar{N}$ is reciprocal. The rate of decrease is exponential. At about $\bar{N} = 300$ the error saturates and no further improvement is achieved. This effect is explained by the right figure where it is observed that the condition number peaks and saturates at about $\bar{N} = 250$. It is interesting to note that even though the condition number is very high between $\bar{N} = 250$ and $\bar{N} = 300$, accuracy is still increasing.

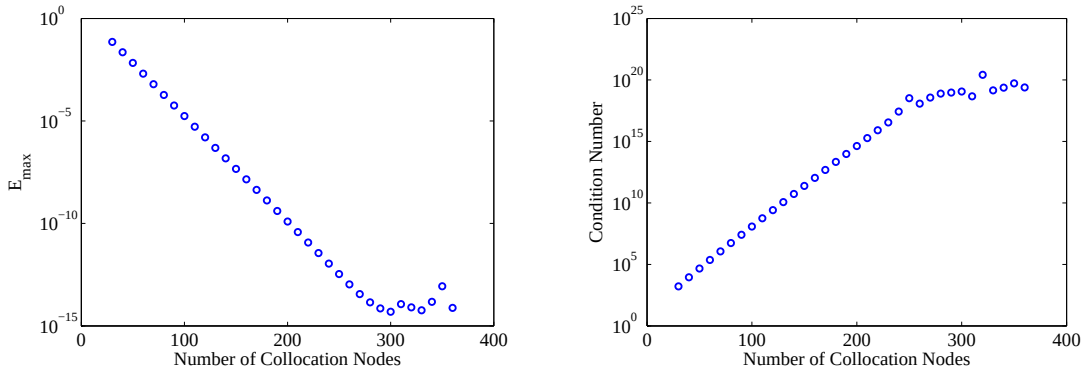


Figure 49: MFS experiment 1: Vary number of collocation points.

4.4.4.4 Experiment 2: Pole Distance

The following experiment extends the previous experiment to include various pole distances. For different number of collocation points $\bar{N} = 30$, $\bar{N} = 60$, $\bar{N} = 120$, $\bar{N} = 240$, and $\bar{N} = 360$ the pole distance is varied and Figure 50 shows the result: Increasing the pole distance from almost zero exponentially decreases the error while the rate is steeper the more collocation points are used. All error curves have a local minimum, i. e., an optimal pole distance, which depends on $\bar{N}$. Further increasing the pole distance leads to more ill-conditioned matrices and higher errors. It is observed again that the best pole distance does not match with the saturation point in the condition number plot. Still, the best operating point is chosen close to the saturation point of the condition number to meliorate the effect of small error perturbations in the equation system.

4.4.4.5 Experiment 3: Non-Uniform Spacing of Collocation Points

In the previous experiments, the node spacing was chosen uniformly. However, nodes extracted from level set functions with sub-pixel accuracy do not have this property. As a result, nodes can be very close to each other, but not further away than the level set grid size. To simulate this situation, the following experiment perturbs every second node of the interface Γ^1 by a certain percentage of the original angle between

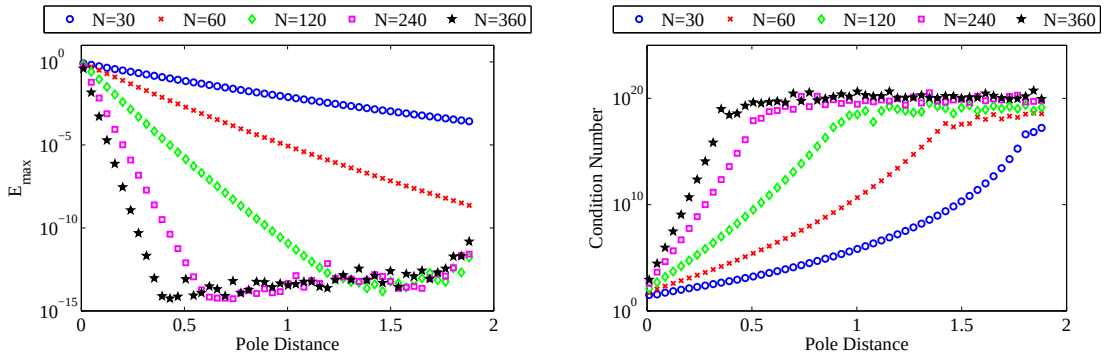


Figure 50: MFS experiment 2: Vary pole distance.

neighboring nodes. At 100% every two nodes coincide and the system becomes singular. As operating points, we choose the saturation point of the condition number from experiment 2, that is $d_p = 1.8$, $d_p = 1.3$, $d_p = 0.9$, $d_p = 0.5$ and $d_p = 0.36$ for the five different $\bar{N}$ from high to low. Figure 51 shows that, while the condition number clearly increases as poles approach each other, the error stays almost the same. The MFS seems, therefore, not very sensitive to non-uniform spacing of the nodes, which is a fortunate discovery with regard to its application in inverse scattering.

4.4.4.6 Conclusions

The above experiments revealed four important properties about the MFS:

1. Accuracy increases with the number of collocation points, but only up to a

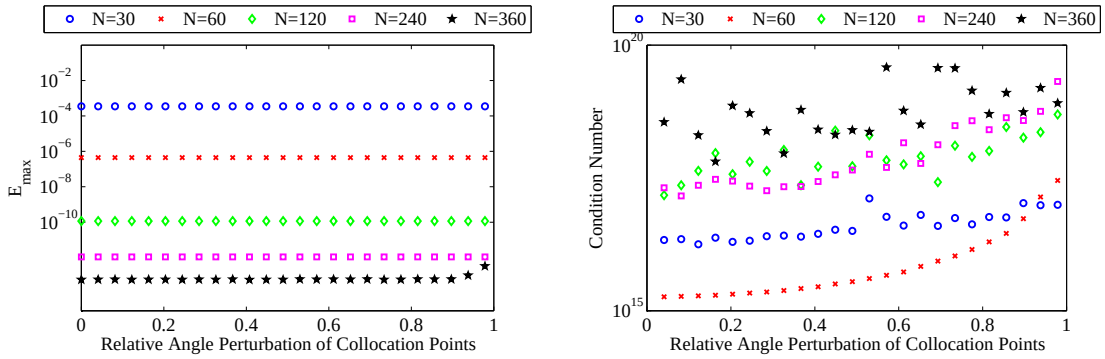


Figure 51: MFS experiment 3: Redistribute collocation points to make spacing between collocation points non-uniform.

certain number of collocation points.

2. Pole distance must be chosen carefully; Too close or too far away distances result in inaccurate results.

3. Non-uniform collocation point spacing seems to have very little influence on the accuracy of the solution.

4. High condition number is not always an indicator for inaccurate solutions.

4.4.5 Formulation for Elastodynamics

In this final section on MFS, the approach for elastodynamic problems is only sketched, since it is analogous to acoustics. We start again with the BVP to be solved

$$T_{ij,j}^{(u^0)} + \gamma^0 u_i^0 = -\sum_{s=1}^{S} \tilde{a}_i^s \delta(x - \tilde{\xi}^s) \quad \text{in } \Omega^0, \tag{178a}$$

$$T_{ij,j}^{(u^1)} + \gamma^1 u_i^1 = 0 \quad \text{in } \Omega^1, \tag{178b}$$

$$T_{ij}^{(u^0)} \mathcal{N}_j^0 = t_i^b \quad \text{on } \bar{\Gamma}, \tag{178c}$$

$$u_i^1 - u_i^0 = 0 \quad \text{on } \Gamma^1, \tag{178d}$$

$$\left((T_{ij}^{(u^1)} - T_{ij}^{(u^0)} \right) \mathcal{N}_j^1 = 0 \quad \text{on } \Gamma^1, \tag{178e}$$

where t_i^b is a generic given function, which prescribes the traction on the outer boundary. For u_i in Eq. (118), $\tilde{a}_i^s \neq 0$, $t_i^b = 0$, whereas for w_i in Eq. (124), $\tilde{a}_i^s = 0$, $t_i^b = dg/du_i$. The numerical solution is again formulated as a linear combination of fundamental solutions (poles), which are located outside of the respective domain,

$$u_i^0(x) = \sum_{n=1}^{P_0} a_j^{0n} U_{ji}^{0*}(x; \xi^{0n}) + \sum_{s=1}^{S} \tilde{a}_j^s U_{ji}^{0*}(x; \tilde{\xi}^s), \tag{179a}$$

$$u_i^1(x) = \sum_{n=1}^{P_1} a_j^{1n} U_{ji}^{1*}(x; \xi^{1n}), \tag{179b}$$

where U_{ji}^{0*}, U_{ji}^{1*} are the fundamental solutions for Ω^0, Ω^1 respectively, ξ^{0n}, ξ^{1n} are chosen pole locations, and a_i^{0n}, a_i^{1n} are unknown pole amplitudes (incorporating both

amplitude and direction of the load). P_0 and P_1 are the number of poles for each domain. Analytical expressions for the fundamental solutions for two-dimensional elastodynamics are given in Appendix C.2.

The goal is again to find the amplitudes a_i^{0n}, a_i^{1n} such that the boundary conditions in Eq. (178) are approximately satisfied as well. For this purpose, the boundary $\bar{\Gamma}$ is discretized by N_0 collocation points x^{0n}, $n = 1, ..., N_0$, and Γ^1 by N_1 collocation points x^{1n}, $n = 1, ..., N_1$, which need not to be sorted. Define the boundary residuals as

$$r_i^1 = T_{ij}^{(u^0)} \mathcal{N}_j^0 - t_i^b \qquad \text{on } \bar{\Gamma}, \tag{180a}$$

$$r_i^2 = u_i^1 - u_i^0 \qquad \text{on } \Gamma^1, \tag{180b}$$

$$r_i^3 = \left((T_{ij}^{(u^1)} - T_{ij}^{(u^0)} \right) \mathcal{N}_j^1 \quad \text{on } \Gamma^1. \tag{180c}$$

The squared norms of the residuals at each collocation point are minimized with respect to the pole amplitudes a_i^{0n}, a_i^{1n}. Combining all unknown amplitudes in one vector, we define

$$\boldsymbol{a} = \left[a_1^{01}, ..., a_1^{0P_0} \,|\, a_2^{01}, ..., a_2^{0P_0} \,|\, a_1^{11}, ..., a_1^{1P_1} \,|\, a_2^{11}, ..., a_2^{1P_1} \right]^T \tag{181}$$

and wish to obtain

$$\boldsymbol{a} = \operatorname{argmin} \left\{ \sum_{n=1}^{N_0} \| r_i^1(x^{0n}) \|^2 + \sum_{n=1}^{N_1} \left(\| r_i^2(x^{1n}) \|^2 + \| r_i^3(x^{1n}) \|^2 \right) \right\}. \tag{182}$$

For simplicity, we do not consider the weighted least squares case here, but it is clear from the derivation in Section 4.4.2 that weighting factors can be incorporated analogously. The right hand side of Eq. (182) can be written equivalently as

$$\sum_{n=1}^{N_0} \| r_i^1(x^{0n}) \|^2 + \sum_{n=1}^{N_1} \left(\| r_i^2(x^{1n}) \|^2 + \| r_i^3(x^{1n}) \|^2 \right) = \| \boldsymbol{A}\boldsymbol{a} - \boldsymbol{b} \|^2, \tag{183}$$

where

$$\boldsymbol{A} = \left[\begin{array}{c|c|c|c} t_{11}^{0*}(\boldsymbol{x}^0;\boldsymbol{\xi}^0) & t_{21}^{0*}(\boldsymbol{x}^0;\boldsymbol{\xi}^0) & 0 & 0 \\ t_{12}^{0*}(\boldsymbol{x}^0;\boldsymbol{\xi}^0) & t_{22}^{0*}(\boldsymbol{x}^0;\boldsymbol{\xi}^0) & 0 & 0 \\ U_{11}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & U_{21}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & -U_{11}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) & -U_{21}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) \\ U_{12}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & U_{22}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & -U_{12}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) & -U_{22}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) \\ t_{11}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & t_{21}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & -t_{11}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) & -t_{21}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) \\ t_{12}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & t_{22}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0) & -t_{12}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) & -t_{22}^{1*}(\boldsymbol{x}^1;\boldsymbol{\xi}^1) \end{array}\right], \tag{184}$$

and

$$\boldsymbol{b} = -\left[\begin{array}{c} t_{11}^{0*}(\boldsymbol{x}^0;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_1 + t_{21}^{0*}(\boldsymbol{x}^0;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_2 - t_1^b(\boldsymbol{x}^0) \\ t_{12}^{0*}(\boldsymbol{x}^0;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_1 + t_{22}^{0*}(\boldsymbol{x}^0;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_2 - t_2^b(\boldsymbol{x}^0) \\ U_{11}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_1 + U_{21}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_2 \\ U_{12}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_1 + U_{22}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_2 \\ t_{11}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_1 + t_{21}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_2 \\ t_{12}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_1 + t_{22}^{0*}(\boldsymbol{x}^1;\tilde{\boldsymbol{\xi}})\tilde{\boldsymbol{a}}_2 \end{array}\right]. \tag{185}$$

The above notation means the following: The term $t_{22}^{0*}(\boldsymbol{x}^1;\boldsymbol{\xi}^0)$, for example, is a matrix of size $N_1 \times P_0$. Its entry (i,j) is the 22-component of the fundamental traction Eq. (364) of domain Ω^0 in the normal direction $\mathcal{N}^1$, caused by the pole at ξ^{0j} and evaluated at the collocation point x^{1i}. Also, $\tilde{\boldsymbol{a}}_1 = [\tilde{a}_1^1, ..., \tilde{a}_1^S]^T$ and $\tilde{\boldsymbol{a}}_2 = [\tilde{a}_2^1, ..., \tilde{a}_2^S]^T$. Finally, the norm $\|\boldsymbol{A}\boldsymbol{a} - \boldsymbol{b}\|^2$ in Eq. (183) is minimized by the pseudo inverse

$$\boldsymbol{a} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}, \tag{186}$$

where standard linear systems solvers can be employed. The numerical solution is now determined and can be evaluated using Eq. (179). The same practical considerations apply as in Section 4.4.3. The main difference to the Helmholtz equation is that the quantities are *vector-valued*, so that each pole and each collocation point is actually assigned n (the number of dimensions) quantities, e.g., the displacement in x_1 and x_2 for two dimensions.

4.5 Experiments

In this section, we present inversion results for the adjoint active contour algorithm derived above. It should be noted that all the experiments presented in this thesis are *synthetic*, meaning that the receiver measurements are obtained by simulating the wave field using the MFS for a chosen and given *true* interface Γ^1_{true}. By using synthetic experiments, we can get insight into what the algorithm can achieve (or not achieve even) in the perfect scenario, where the wave propagation model is able to exactly describe the measurements. Keep in mind that we deal with an extremely ill-posed problem where the amount of information (finite set of receiver measurements) is far less than the number of unknowns (curve, an infinite dimensional object). Therefore, even in the perfect case, we cannot always expect perfect reconstruction of the true interface, since non-uniqueness of the minimizer, local minima or numerical inaccuracies in the forward solution may prevent success of the method.

Our experiments are, therefore, considered proof-of-concept, and application of this method to real data will require an additional level of understanding and consideration for the real problem at hand, where model errors and measurement noise may adversely affect the reconstruction. With this goal of real applications in mind, however, this section presents a series of analyses on synthetic data that will provide important practical guidelines for future applications.

First, results for acoustics (Helmholtz equation) are presented in Section 4.5.1. This case is used to qualitatively analyze the inversion algorithm and to extend the method by frequency hopping, multiple source approaches and the translation-scale approach, which are important extensions to improve the chance of successful reconstruction in general. In Section 4.5.2, these extensions are employed to attempt inversion of problems in elastodynamics.

4.5.1 Acoustics

4.5.1.1 Proof of Concept

In this section, a result from a MatLab implementation of the adjoint active contour inversion algorithm is shown for a simple geometry using only few receivers and sources. The wave numbers for domains Ω^0 and Ω^1 are $k_0 = 1/\sqrt{100}\,\text{rad/m}$ and $k_1 = 1/\sqrt{10}\,\text{rad/m}$, respectively, and a Robin boundary condition with $\lambda = 0.5$ is applied on the surface boundary $\bar{\Gamma}$. Starting from an initial guess different than the true shape, the algorithm is expected to converge back to the true shape. In practice, the goal is to use as few sources and receivers as possible, since these are important cost factors. Figure 52 and Figure 53 show a successful inversion result recovering an ellipse using two sources and six receivers. The energy plot reveals that recovering the global position of the inclusion decreases energy rapidly. Then, more detailed features of the object shape are obtained in a phase with slower energy drop rate.

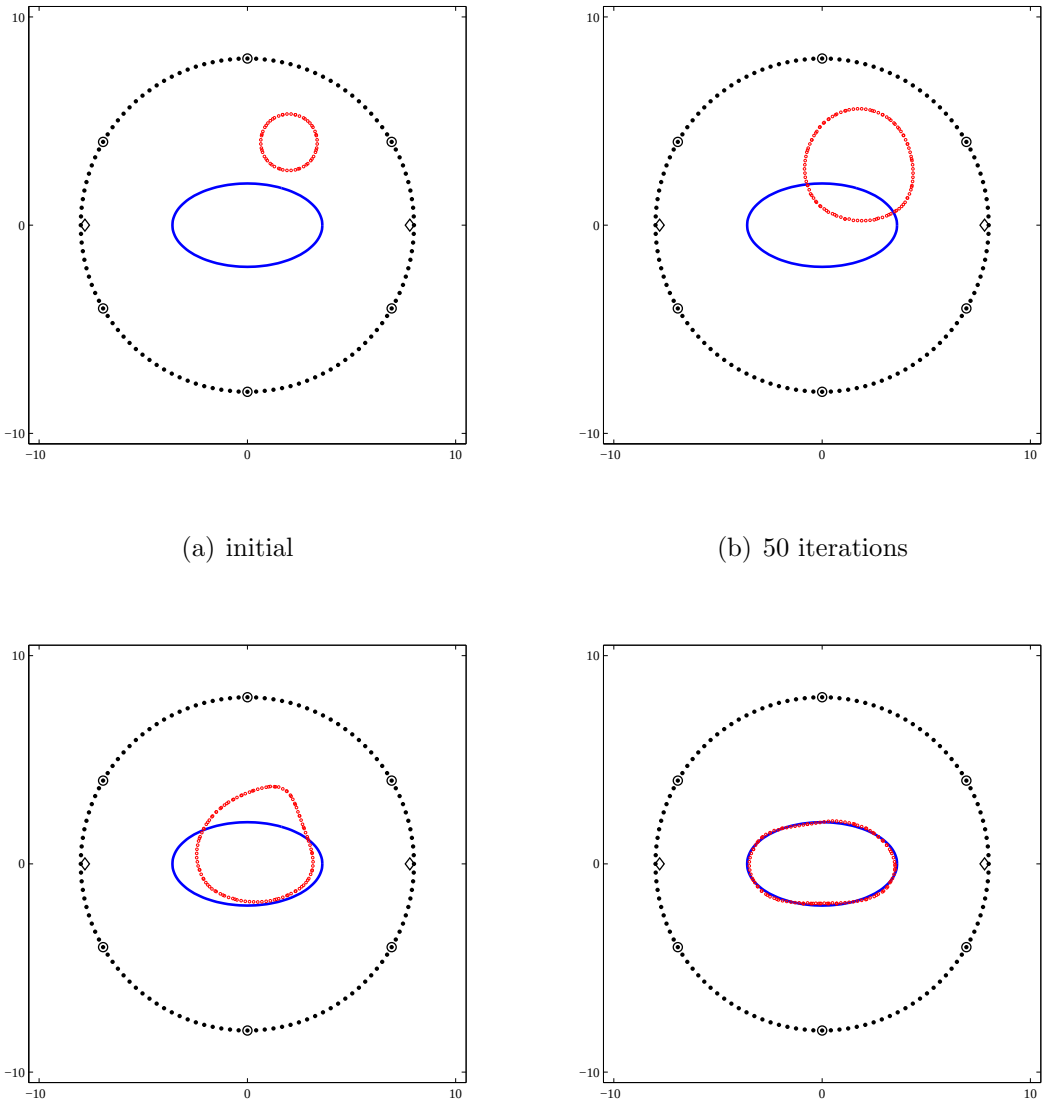
Even though this experiment might appear to be a simple task because of the regularity of the shapes, we point out that our algorithm does not assume such simple shapes, but operates in the space of two-dimensional curves, which makes the problem highly ill-posed considering that the amount of measured data is limited to a small, finite set of values. This result is considered proof-of-concept for adjoint active contours, and more careful analysis towards practical considerations will be performed in the following sections.

In this initial testing phase using a MatLab implementation, several observations were made that strongly influenced the later course of this research:

1. Even though the energy is consistently minimized, the chance of recovering the true shape is fairly low.
2. Accurate reconstruction requires careful parameter tuning (receiver locations, source locations, frequency) depending on the choice of initial interface, true

interface, material properties, etc.

These experiences have made the direct application of the method to real data unrealistic at this stage of the research, since without the knowledge of the true interface and according parameter tuning, successful reconstruction is not expected, not to mention the negative effect of model errors and noise when dealing with real data. In order to



(a) initial (b) 50 iterations

(c) 100 iterations (d) final (450 iterations)

Figure 52: Inversion of an ellipse for acoustics. Outer boundary (black), current shape (red), true shape (blue), 2 sources (black diamond), 6 receivers (black circles).

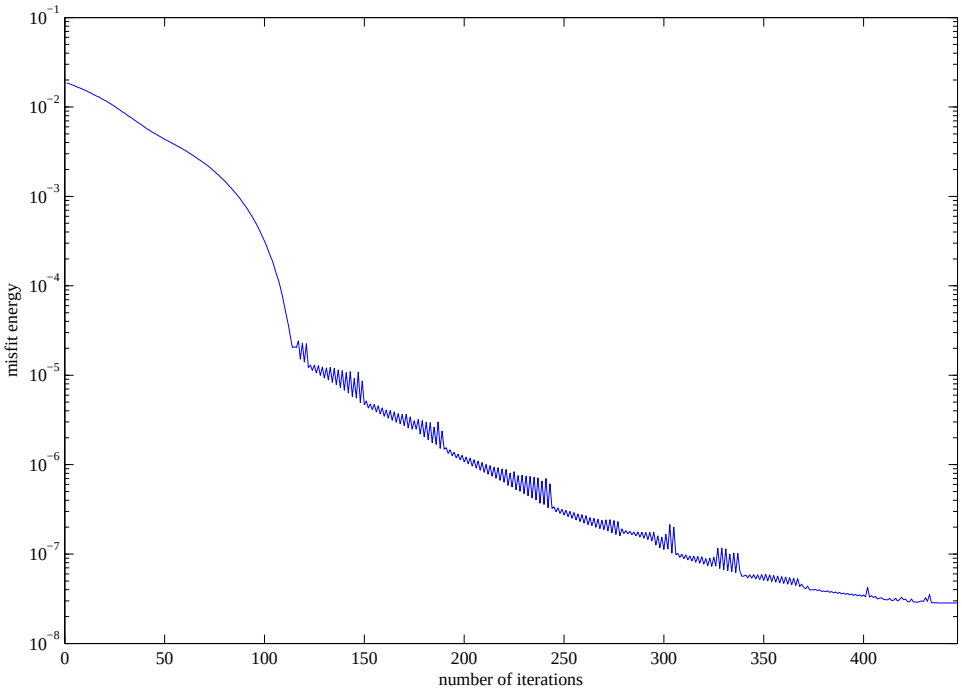


Figure 53: Energy plot for inversion result in Figure 52.

better understand the method and these difficulties, we restrict ourselves to synthetic examples and draw general conclusions from the parameter tuning experience that will help towards the goal of applying this method to real-world problems.

4.5.1.2 Testing Environment

Towards the goal of interactively exploring the influence of different parameters, we reimplement the algorithm in C++ and integrate it with the software *PDE Toolkit* developed by Vintinura Imaging Inc. The PDE Toolkit provides a modular structure for contour evolution as well as user interaction through a TCL/TK graphical user interface. This software framework provides a much more flexible environment to interact with the inversion algorithm, such as creating arbitrary hand-drawn setups, stopping the program and changing parameters on the fly, interactively modifying the current contour, etc.

With the help of this implementation, we will investigate several properties of the adjoint active contour method using an acoustic problem setup. Acoustics is chosen for this investigation because its lower computational as well as conceptual complexity makes it more suitable for interactive exploration. Since both acoustics and elastodynamics represent wave propagation problems, we expect that the conclusions drawn from acoustics will translate to elastodynamics in a similar way. Experiments for elastodynamics are shown at the very end of this thesis.

The test setup is chosen as shown in Figure 54. The grid represents a square with an edge length of 20 meters, and is discretized (for the purpose of level set evolution) by 101 times 101 pixels. The wave speeds are $c_0 = 340\,\mathrm{m/s}$ (similar to air) for the ambient domain Ω^0 and $c_1 = 1450\,\mathrm{m/s}$ (similar to water) for the inclusion domain Ω^1, and a Robin boundary condition with $\lambda = 0.5$ is applied on the surface boundary $\bar{\Gamma}$. All the geometries are hand-drawn and thus represent arbitrary shapes. Initial tests on concave inclusion shapes have proven very difficult compared to convex

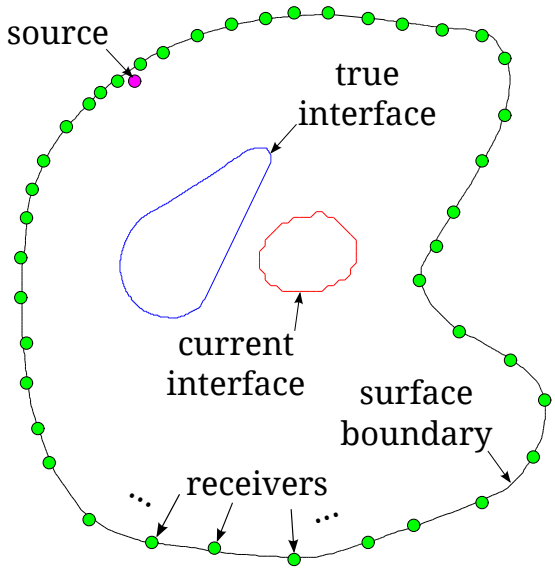


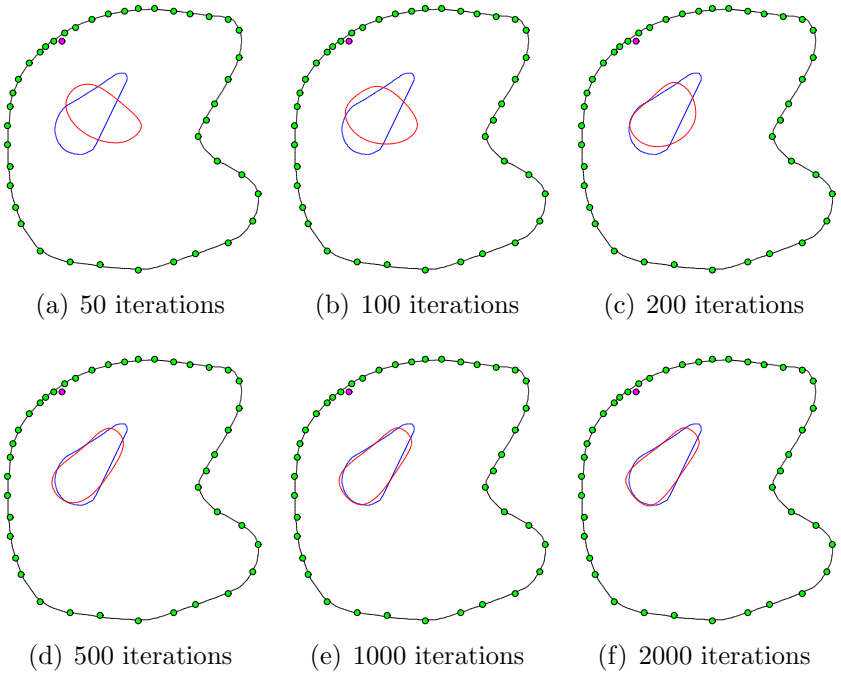
Figure 54: Setup for further analysis of the adjoint active contour method for acoustics. The spatial dimensions are 20 meters by 20 meters, the wave speeds are $c_0 = 340$ m/s for the ambient domain Ω^0 and $c_1 = 1450\,\mathrm{m/s}$ for the inclusion domain Ω^1.

shapes, which is why we choose an inclusion shape that is only slightly concave on the north-west part. Moreover, the inclusion shape features a corner with relatively high curvature (north-east), an exact straight line (south-east) and a curve segment with moderate curvature (south-west). The surface boundary is clearly concave. The large number and complete distribution of receivers represents a rather ideal scenario to ensure that we do not attempt an impossibly ill-posed problem. We will investigate the consequences of lowering the number of receivers in Section 4.5.1.5. We choose only one source for this setup, and will study the effect of several sources in Section 4.5.1.4.

4.5.1.3 Approaches for Multiple Frequencies

The adjoint active contour method was derived under the assumption of a harmonic source term with constant frequency ω. Initial experiments suggest that the use of a single frequency leads to very limited reconstruction results. This observation is corroborated by applying the algorithm to the test setup for different frequencies. Whereas the reconstruction for a very low frequency of $\omega = 50\,\mathrm{rad/s}$ (the corresponding wave length is 42.7 m in air and 182.2 m in water) in Figure 55 is not a complete failure (even though it takes a large amount of iterations), the final results for other frequencies in Figure 56 are less satisfying. The algorithm is stopped for these experiments when the interface gets close to the outer boundary or after 1000 iterations.

From energy plots, it becomes clear that these unsatisfying results are due to local or non-unique global minima, from which the energy method cannot escape. On the one hand, it confirms what a pessimistic researcher would expect: Variational methods guarantee convergence to the global minimum only if the current estimate is "close enough" to the global minimizer, so that there are no other attracting local minima. Hence, the presented method cannot guarantee to obtain the global minimizer, and



(a) 50 iterations (b) 100 iterations (c) 200 iterations

(d) 500 iterations (e) 1000 iterations (f) 2000 iterations

Figure 55: Inversion process using $\omega = 50\,\mathrm{rad/s}$.

the experiments show that, in fact, the method usually gets stuck in local minima when only one frequency is used. On the other hand, there are promising signs for success when using multiple frequencies as is explained next.

From an energy point of view, the topology of the energy is defined by the chosen frequency. How exactly frequency changes the topology of the energy is a difficult question, but intuitively it is reasonable to believe that there is a high probability that the local minima of energies resulting from different frequencies do usually not coincide. When using multiple frequencies, we therefore have several energies available which describe the same problem and between which we can switch as needed. This idea leads to the concept of *frequency hopping*, where we switch the driving energy's frequency under certain conditions. Automatic switching rules are, for example, to switch when one energy has converged or after a certain number of iterations. Early experiments suggest that this approach is the most promising and important extension to the one-frequency adjoint active contour approach presented in this chapter.

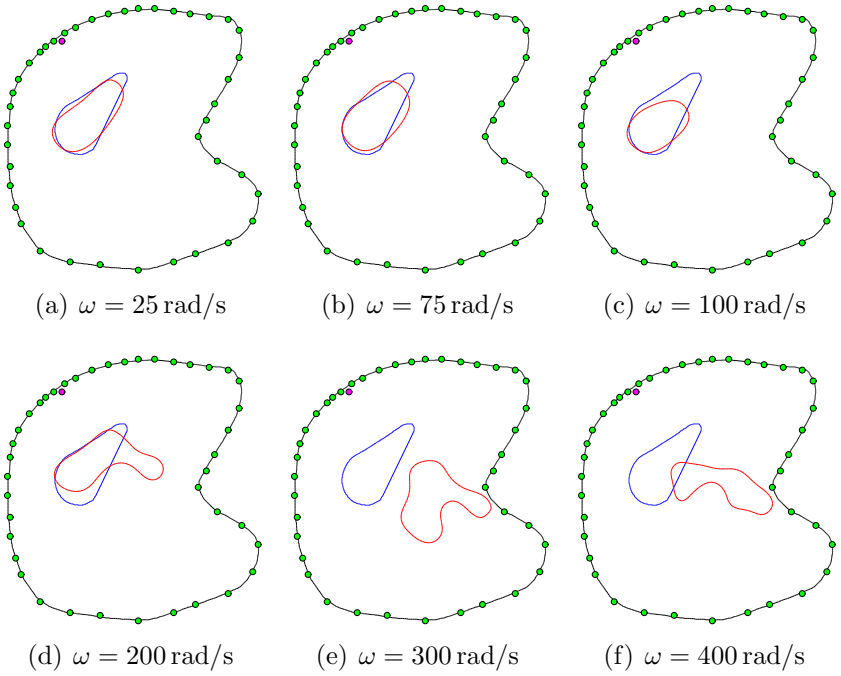
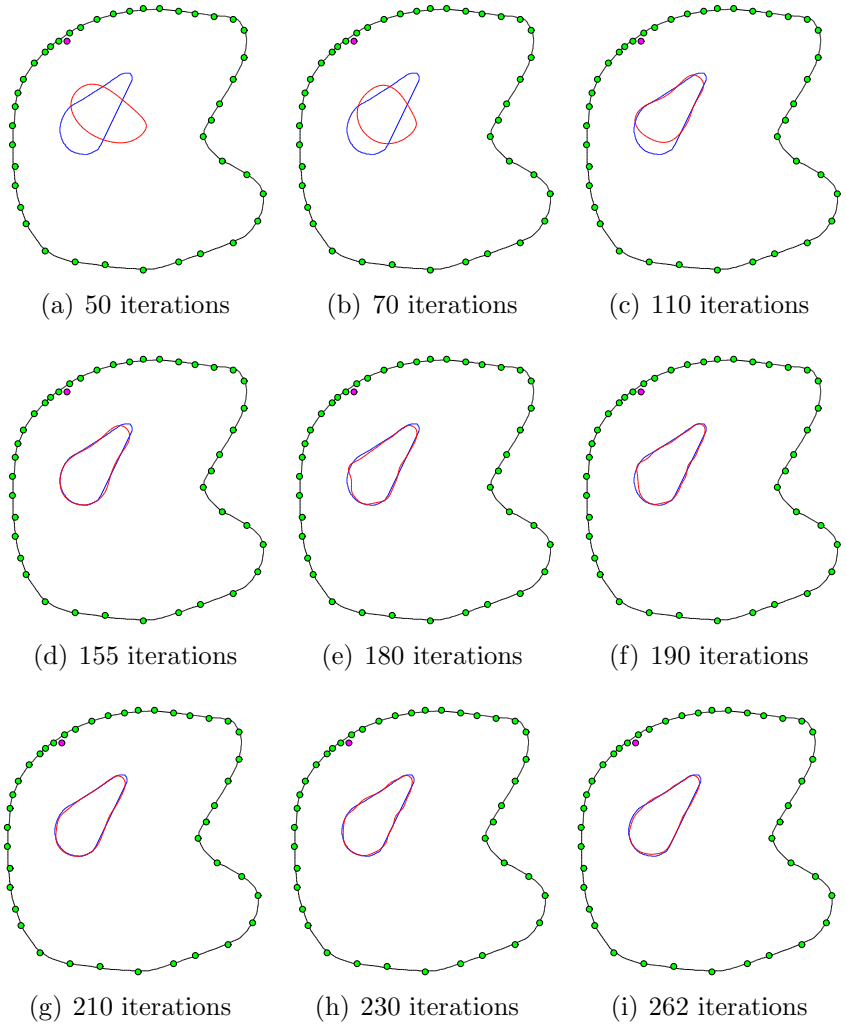


Figure 56: Final inversion results for different constant frequencies.

With the help of the graphical user interface, we present a proof-of-concept result for frequency hopping in Figure 57, where we choose the switch time and new frequency manually in response to the current performance of the algorithm. That is, if a certain frequency seems to perform well, we keep it; if a frequency gets stuck or adversely changes the contour, we switch to another frequency. Experience from such experiments supports the following approach: Lower frequencies correspond to high wave lengths, and are, therefore, more useful for recovering macroscopic properties such as global location and size of the inclusion, whereas higher frequencies result in low wave lengths, which are capable of resolving finer details of the shape. Smaller frequencies, for example, would usually be able to recover circular or slightly elliptical shapes, but fail for more complicated geometries. With a good initial guess, however, higher frequencies are able to recover high-curvature curve segments, but they are incapable of capturing global location and size. In general, each frequency works to achieve its goal, but tends to destroy some of the good work done by other frequencies. Overall, however, chances to converge to the basin of attraction of the



(a) 50 iterations (b) 70 iterations (c) 110 iterations

(d) 155 iterations (e) 180 iterations (f) 190 iterations

(g) 210 iterations (h) 230 iterations (i) 262 iterations

Figure 57: Manual frequency hopping result.

global minimum is increased with frequency hopping because the true shape is the global minimizer for all frequencies.

The result of manual frequency hopping is shown in Figure 57 and the according frequency plot is Figure 58. First of all, the final result Figure 57(i) proves that frequency can be useful to achieve more accurate results with less iterations. We can divide the approach in roughly four phases as shown in Figure 58. Phase 1 (iterations 1 to 70) is characterized by low frequency, which achieves to center the curve and to obtain a rough estimate of its size. Phase 2 (iterations 71 to 145) is characterized by mid-level frequencies and works efficiently to recover the majority of the shape in less

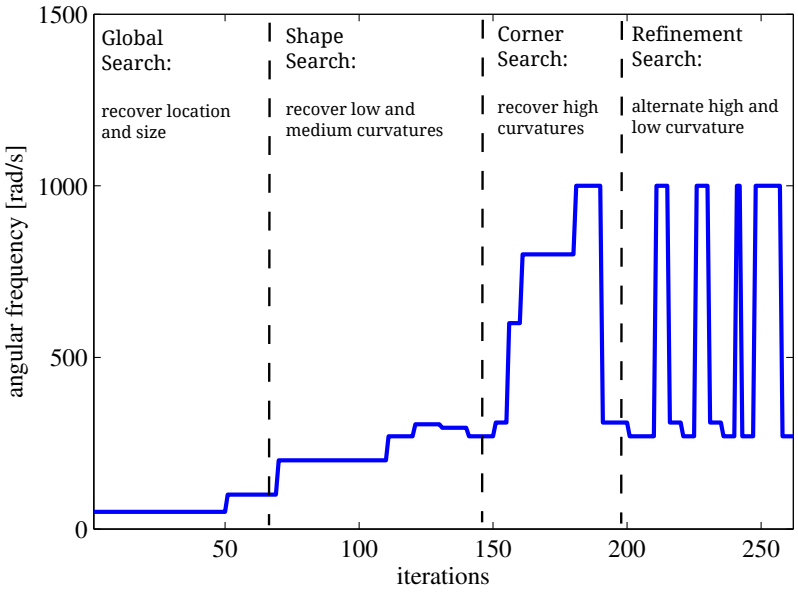


Figure 58: Manual frequency hopping: Frequencies over computation time and corresponding search phases.

than 100 iterations (see Figure 57(d)). Phase 3 (iterations 146 to 200) is characterized by high frequencies and obtains a more accurate estimate of the high-curvature corner in the north-east part of the curve (see Figure 57(f)). However, the high frequency phase destroys some of the mid-level-curvature reconstruction, most obviously seen in the south-west part of the curve in Figure 57(f). Running the contour for longer time in the high-frequency regime often results in undesired wiggling of the curve, so this phase must be approached with extreme care. In the last phase 4 (iterations 201 to 262), we attempt to refine the recovery by alternating between high and mid-level frequencies. Each alternation refines some of the properties destroyed in the previous alternation (e.g., high frequency destroys reconstructed low curvature; low frequency destroys reconstructed high curvature), but fortunately, this quick frequency switching has an overall improving effect, and the contour stays attracted to the global minimum.

It should be noted that this manual frequency hopping result is obtained under substantial manual effort where we manipulate frequency according to the current

shape evolution and knowledge of the true shape. It is, therefore, very ambitious to design a fully automatic inversion algorithm that would work on a large class of problems. The following extension to frequency hopping could be considered in the future when a fully automatic inversion algorithm is desired. Assume that measurements are available for different frequencies $\omega_1, ..., \omega_K$. We denote the measurement for frequency ω_k at receiver location x_r as $\hat{u}(x_r; \omega_k)$. Then one can minimize the weighted sum of the misfit terms for each frequency, i.e.,

$$E_{\text{joint}}(\Gamma^1, ..., \Gamma^L) = \sum_{k=1}^{K} \alpha_k \frac{1}{2} \sum_{r=1}^{R} \|u(x_r; \omega_k) - \hat{u}(x_r; \omega_k)\|^2. \tag{187}$$

This joint frequency energy is straightforward to minimize since each term represents a misfit term derived in this thesis, so that

$$\nabla_{\Gamma^l} E_{\text{joint}}(\Gamma^1, ..., \Gamma^L) = \sum_{k=1}^{K} \alpha_k \nabla_{\Gamma^l} E(\Gamma^1, ..., \Gamma^L; \omega_k), \tag{188}$$

where

$$E(\Gamma^1, ..., \Gamma^L; \omega_k) = \frac{1}{2} \sum_{r=1}^{R} \|u(x_r; \omega_k) - \hat{u}(x_r; \omega_k)\|^2, \tag{189}$$

which is the standard energy for $\omega = \omega_k$. We call this method the *joint frequency approach*. Observe that—if the parameters α_k are made computation-time-dependent—one can obtain frequency hopping by setting all α_k to zero except for the active frequency. More generally, if more than one weight is non-zero, the weights α_k have two purposes: First, they should be chosen to normalize the effect of different orders of wave amplitudes at different frequencies. Second, they need to account for the different search phases indicated in the manual approach above. Typically, one would chose higher weights for lower frequencies initially and then shift the weights to higher frequencies over computation time. Finding efficient ways to determine optimal frequencies and weights will be a crucial objective in future efforts.

4.5.1.4 Approaches for Multiple Sources

Whereas multiple frequencies are required to account for different scales of the problem, multiple sources may be necessary to obtain sufficient information from all parts of the imaged region. If a region is reached by waves only via reflections, it is likely that recovery of this part of the domain is not very accurate. This effect is referred to as *lack of illumination*. By placing multiple sources across the surface boundary, illumination can be improved and better inversion results are expected. For now, we assume one frequency, so that for multiple sources $s_1, ..., s_S$ and multiple receivers $r_1, ..., r_R$, we have a matrix $\hat{U}(i,j) \in]mathbbC^{R \times S}$, which contains the measured value at r_i when only source s_j is active. We consider three inversion approaches to make use of this data.

The first approach is the *additive source approach*, where the simultaneous (i.e., additive) action of all sources is considered, i.e., $\hat{u}(x_r) = \sum_{j=1}^{S} \hat{U}(r,j)$. Due to linearity, the sum of single-source solutions is the solution for the source being the sum of the single sources. Hence, we can minimize the standard energy using the sum of the measured data. The simulated wave field is obtained by solving the forward model using the sum of the single sources as the excitation term. The MFS handles this situation very easily. Observe that we could not take this approach for the multiple frequency case, because the sum of two solutions with different frequencies does not solve the original harmonic PDEs, and therefore, our minimization approach does not apply.

The second and third approach are analogous to the multiple frequency approaches. In the *joint source approach*, we minimize the energy

$$E_{\text{joint}}(\Gamma^1, ..., \Gamma^L) = \sum_{j=1}^{S} \alpha_s \frac{1}{2} \sum_{i=1}^{R} \|U(i,j) - \hat{U}(i,j)\|^2. \tag{190}$$

where U is the matrix of simulated field values for each source-receiver pair analogous to $\hat{U}$. Since there is no apparent reason to favor one source over the other, $\alpha_s = 1 \, \forall s$

in the following. The joint energy is again minimized by the sum of its gradients

$$\nabla_{\Gamma^l} E_{\text{joint}}(\Gamma^1, ..., \Gamma^L) = \sum_{j=1}^{S} \alpha_s \nabla_{\Gamma^l} E(\Gamma^1, ..., \Gamma^L; s_j), \tag{191}$$

where

$$E(\Gamma^1, ..., \Gamma^L; s_j) = \frac{1}{2} \sum_{r=1}^{R} \|u(x_r; s_j) - \hat{u}(x_r; s_j)\|^2. \tag{192}$$

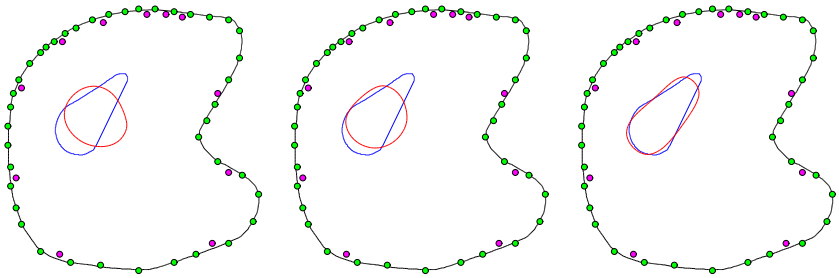
The third approach is *source hopping* where we pick one single source (and its corresponding receiver measurements), minimize, switch to another single source, minimize, etc. For automatic hopping, the active source is switched after a few fixed iterations, unless there is reason to believe that one source is more effective than another.

All these approaches make use of information from multiple sources. Regarding efficiency, hopping is the fastest per iteration, followed by additive and joint. Hopping has the same complexity as the single-source method. The additive approach is only very slightly slower per iteration because one needs to sum over the sources in each iteration. The joint approach is clearly slower than the other two approaches, because S different forward and adjoint solutions need to be computed. Luckily, only the right-hand-side of the system matrix in the MFS changes, so that solutions for all S right-hand-side vectors can be obtained from the same linear system solving procedure, which prevents complexity from being S times larger. These statements, however, refer to the per-iteration complexity. What really matters is the efficiency to recover shape, and experiments will show that the answer is indeed not as easy as comparing per-iteration computation cost. The following paragraphs aim at shedding light onto the strengths and weaknesses of each approach.

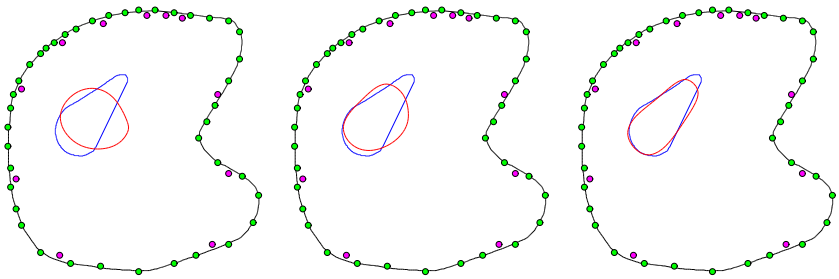
First, we add 10 sources to the test setup of Section 4.5.1.2 and consider again the low frequency case $\omega = 50\,\text{rad/s}$. Each multi-source approach is run to convergence and the result is shown in Figure 59. All approaches capture the rough shape but miss the high-curvature segment in the north-east of the curve. The final result does

not provide considerable improvement over the single-source case in Figure 55, which may be surprising at first. It shows, however, that frequency is a far more important parameter than the source location and even number of sources. In the first phase of frequency hopping (global search), it seems, therefore, advisable to choose the method with the lowest computational cost, which is either the additive or hopping approach. Inspecting the source hopping result in Figure 59, it seems that it converges slower than the additive approach, so that the additive approach is recommended for low frequencies.

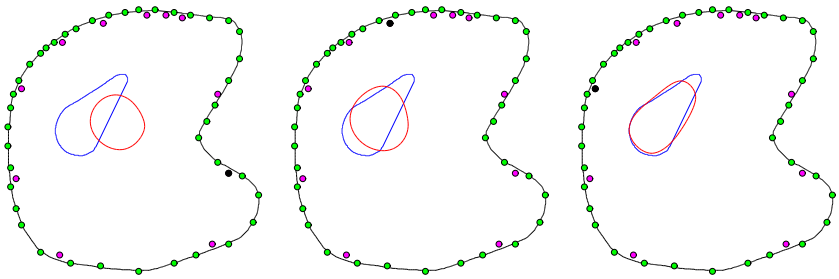
Now, consider a mid-level frequency case with $\omega = 200\,\mathrm{rad/s}$, the result of which is



(a) additive approach at 50, 100 and 1000 iterations

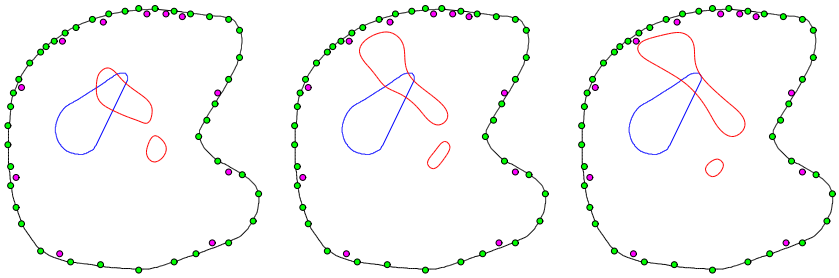


(b) joint approach at 50, 100 and 1000 iterations

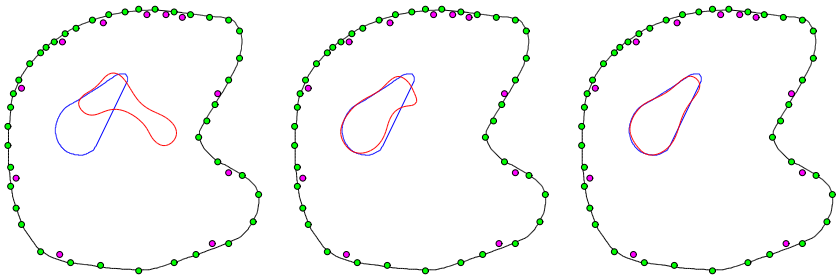


(c) hopping approach at 50, 100 and 1000 iterations

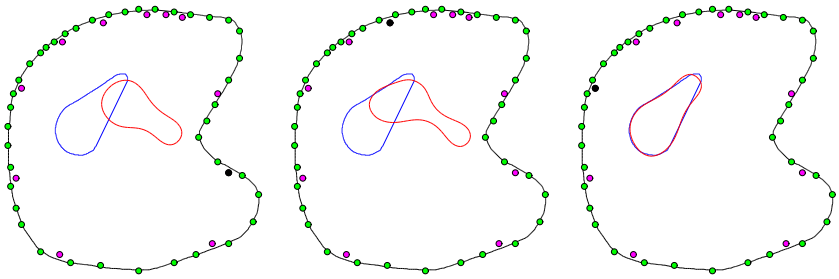
Figure 59: Different multi-source approaches at frequency $\omega = 50\,\mathrm{rad/s}$.



(a) additive approach at 50, 100 and 180 iterations



(b) joint approach at 50, 100 and 1000 iterations



(c) hopping approach at 50, 100 and 2000 iterations

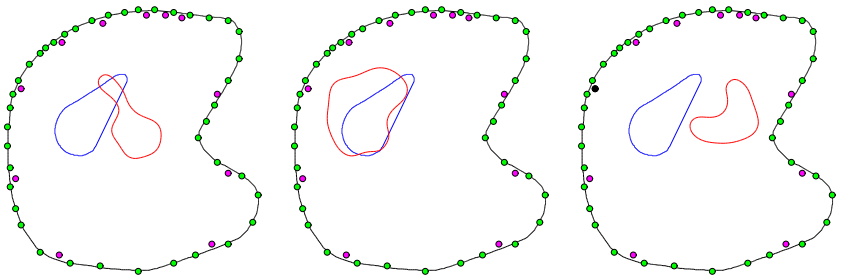
Figure 60: Different multi-source approaches at frequency $\omega = 200\,\mathrm{rad/s}$.

shown in Figure 60. Clearly, the additive approach fails badly for this case. However, both the joint and the hopping approach converge to fairly accurate reconstructions with only small errors in the high-curvature segment. It is reasonable to expect that the additive approach will perform worse than the other two approaches, in general, since we essentially only take R measurements by summing the receiver responses over all sources, whereas for the other two approaches, an energy term is involved for each receiver-source pair, i.e., S times R measurements are used. From this argumentation, one would expect the additive approach to be inferior. The experiment for low frequencies above showed that this argument does not apply to

low frequencies, but it is corroborated in this experiment for $\omega = 200\,\mathrm{rad/s}$. The question remains if the joint or the hopping approach is faster. From Figure 60 it seems obvious that the joint approach converges faster, but one has to keep in mind that the joint approach takes more time per iteration than the hopping approach. The joint approach at iteration 50 is similar to the hopping approach at iteration 100, but it is observed in the experiment that computation time per iteration is only ca. 1.5 times higher for the joint approach, so the joint approach is preferred in this scenario.

For a higher frequency of $\omega = 400\,\mathrm{rad/s}$, all approaches fail leading to various inaccurate results shown in Figure 61, where the joint approach leads to the least bad reconstruction. This experiment confirms the claim that higher frequencies are more suitable to recover details of shape (higher curvature), and it is not expected that a contour with an initial guess far from the true shape will converge for high frequencies. Thus, it is more sensible to analyze the high-frequency for cases where the contour is already close to the true shape as, for example, in the final results in rows 2 and 3 of Figure 60.

For analyzing higher frequencies, we thus resort again to the frequency hopping approach, where we try to recover a good shape estimate using low and mid-level frequencies before applying higher frequencies for refinement. As explained above, we use the additive approach and low frequencies to recover size and location in phase 1



(a) additive approach (b) joint approach (c) hopping approach

Figure 61: Final results after 1000 iterations for $\omega = 400\,\mathrm{rad/s}$.

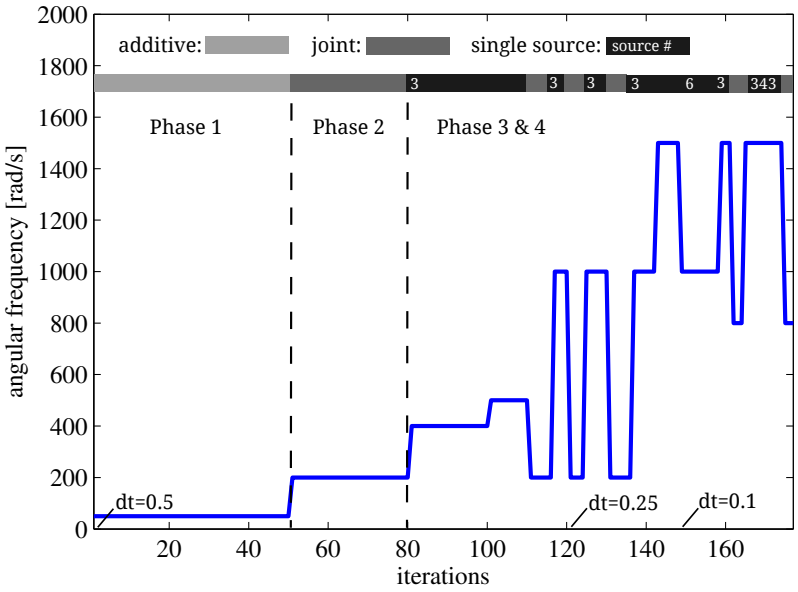


Figure 62: Manual frequency hopping and multiple source approaches to obtain inversion result in Figure 63.

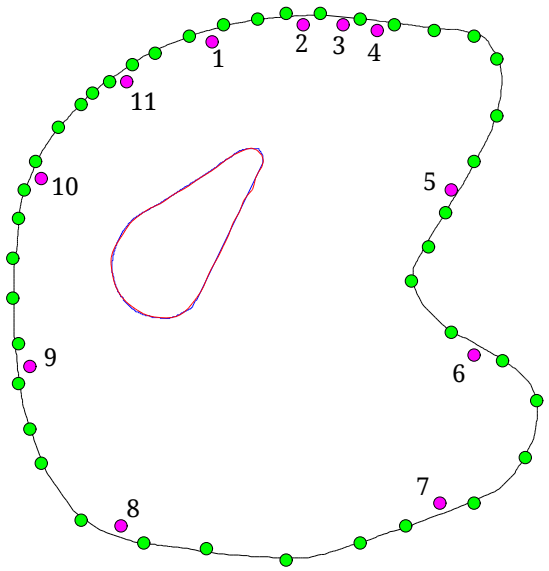


Figure 63: Final result for manual frequency hopping and multiple source approaches after 180 iterations.

as shown in Figure 62. Phase 2 is characterized by mid-level frequency using the joint source approach. In phase 3, careful switching between the joint source approach and single active sources achieves the final reconstruction result in Figure 63. Choosing a single source at a time at high frequency has the effect that high curvature is recovered

at the curve segment closest to the active source with relatively low damage to the rest of the curve. Since in this example, the north-east segment of the curve has the highest curvature, we most often activate source number 3 (labels for the sources are given in Figure 63). This approach can be viewed as manual source hopping where the order sequence is chosen by the user. The reconstruction in Figure 63 is very accurate, which is attributed to using multiple sources and multiple frequencies in combination. Moreover, tuning of the time step dt is applied in this experiment: As long as the contour is far away from the global minimum, a large time step is beneficial since it speeds up the curve evolution and may even help to escape small local minima. As the contour approaches the true curve, however, one should decrease the time step to avoid jumping beyond the global minimum in the gradient descent method.

In summary, when multiple sources and frequencies are available, the experiments above justify the following approach: Apply the four-phase (global, shape, corner, refinement) frequency hopping as presented in Section 4.5.1.3, where the additive source method is most useful for the first (low-frequency) phase, the joint source approach is preferred for the second (mid-level frequency) phase, and a careful combination of joint source approach and source hopping approach is chosen in the third (high frequency) and fourth (mid and high frequency) phase. In the following, we will consider more experiments of the same type which analyze the effect of changing the number and location of sources and receivers.

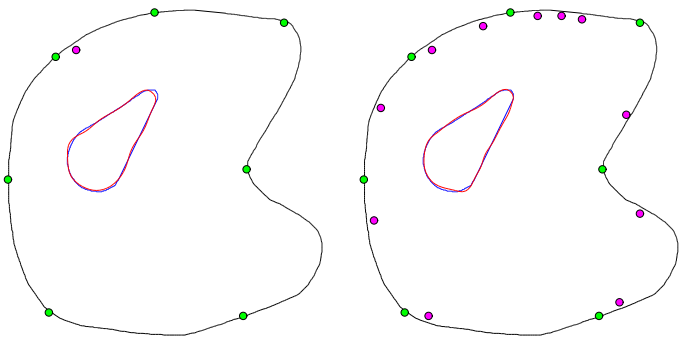
4.5.1.5 Sources and Receivers

In this section, the effect of number and location of sources and receivers is studied. We continue to manually apply frequency hopping, multiple-source approaches and adjustment of the time step as these extensions are key to success. Above, we have shown two successful results, first for 1 source and 41 receivers (Section 4.5.1.3

frequency hopping), and second, for 11 sources and 41 receivers (Section 4.5.1.4 frequency hopping plus multi-source approaches), and we observed that the increase in sources improved accuracy and robustness of the inversion. To complete this study, we consider the same experiments with only 7 receivers, which are placed with approximately even distance along the surface boundary.

The successful results in Figure 64 are obtained again using careful manual frequency hopping and multi-source approaches (in the second figure only, since the first experiment has only one source). We thus conclude that the adjoint active contour method has the potential to reconstruct shape from only very few measurements and sources if carried out carefully.

In the previous experiments, receivers and sources were distributed along the whole circumference of the surface boundary. This type of excitation and measurement setup is reasonable to assume in medical or non-destructive testing scenarios where one has access to the whole surface of the object to be imaged. However, scenarios can be imagined where one does not have access to all parts of the boundary so that receivers and sources need to be placed along a certain segment of the outer boundary. This type of problem is also relevant when considering half-space domains, for example for seismic or sonar imaging, where parts of the object boundary are not directly



(a) 1 source, 7 receivers (b) 11 sources, 7 receivers

Figure 64: Final results using manual frequency hopping and multi-source approaches in (b) for only 7 receivers.

exposed to sources or receivers, but can only be reached via boundary reflections or via transmission through the object itself. We consider such a scenario in Figure 65 for the test setup in Section 4.5.1.2 with 11 sources and 7 receivers distributed along the north-west part of the surface boundary. We refer to this experiment as the "zone" experiment, since sources and receivers can only be placed in a certain restricted zone.

The result for the global search using a low frequency of $\omega = 50\,\mathrm{rad/s}$ and the joint source approach (to ensure we use the most accurate approach) is shown in Figure 66, and is not very encouraging because it takes many iterations to reach a good guess for the location and size of the inclusion. This phenomenon is referred to as *lack of illumination*: The curve segment farthest away from the source-receiver zone is neither excited nor measured directly but only via boundary and interface effects, which results in a low sensitivity on this part of the curve. One approach to improve this situation is to consider global translation and scale of the curve to avoid certain curve segments not to be updated. The two concepts are explained next.

Translation of a contour can be written as a curve flow with a constant force vector $\bar{\mathcal{F}}$ with unit norm, i.e., $\Gamma_t = \bar{\mathcal{F}}$ where $\bar{\mathcal{F}}$ does not depend on x and $\|\bar{\mathcal{F}}\| = 1$.

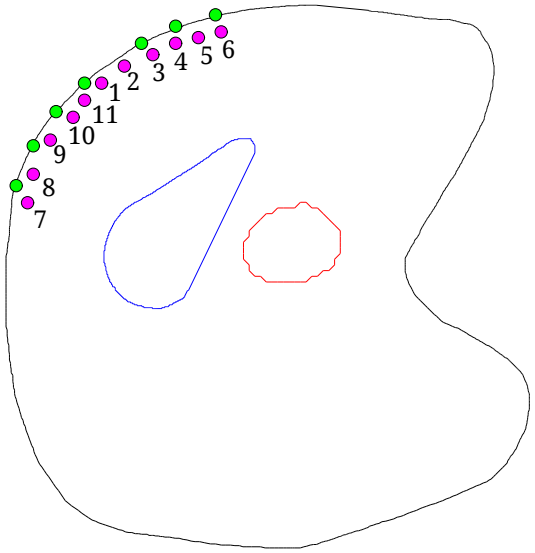
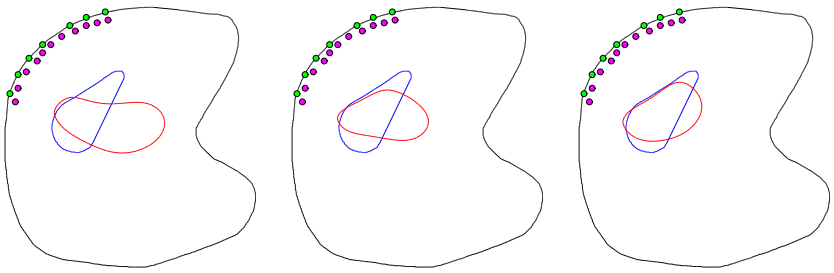


Figure 65: Initial setup for the zone experiment.



(a) iteration 100 (b) iteration 500 (c) iteration 1000

Figure 66: Global search phase result for zone experiment for $\omega = 50\,\text{rad/s}$ and the joint source approach.

Our goal is to find the optimal $\bar{\mathcal{F}}$ to decrease the data misfit energy in the gradient direction. First, not that

$$\Gamma_t = (\bar{\mathcal{F}} \cdot \mathcal{N})\mathcal{N} + (\bar{\mathcal{F}} \cdot \mathcal{T})\,\mathcal{T} = (\bar{\mathcal{F}} \cdot \mathcal{N})\,\mathcal{N}, \tag{193}$$

since infinitesimal changes in the tangential component do not change the curve. We have, therefore, that

$$\frac{dE}{dt} = -\int_\Gamma (F\mathcal{N})\Gamma_t \, d\Gamma = -\int_\Gamma (F\mathcal{N})(\bar{\mathcal{F}} \cdot \mathcal{N})\mathcal{N} \, d\Gamma = -\int_\Gamma F(\bar{\mathcal{F}} \cdot \mathcal{N}) \, d\Gamma, \tag{194}$$

where F is the adjoint active contour force term defined along the curve. Minimizing dE/dt with respect to $\bar{\mathcal{F}}$ under the unit norm constraint yields

$$-\int_\Gamma F\mathcal{N} \, d\Gamma + \lambda 2\bar{\mathcal{F}} = 0, \tag{195}$$

where λ is a Lagrange multiplier and not to be confused with the parameter in the Robin boundary condition for acoustics. Choosing $\lambda = \| \int_\Gamma F\mathcal{N} \, d\Gamma\|/2$ to satisfy the unit norm constraint, one obtains

$$\bar{\mathcal{F}} = \frac{\int_\Gamma F\mathcal{N} \, d\Gamma}{\| \int_\Gamma F\mathcal{N} \, d\Gamma\|}. \tag{196}$$

Note that the unit norm constraint is arbitrary in the sense that we could have chosen any constant since the time step will be adapted by the same factor to make the curve flow numerically stable. So, in practice, we only need to compute $\bar{\mathcal{F}} = \int_\Gamma F\mathcal{N} \, d\Gamma$.

Scaling of a contour can be achieved by either eroding or dilating the curve, i.e., $\Gamma_t = \bar{F}\mathcal{N}$ where $\bar{F}$ is either 1 or -1. With this constraint, the energy derivative becomes

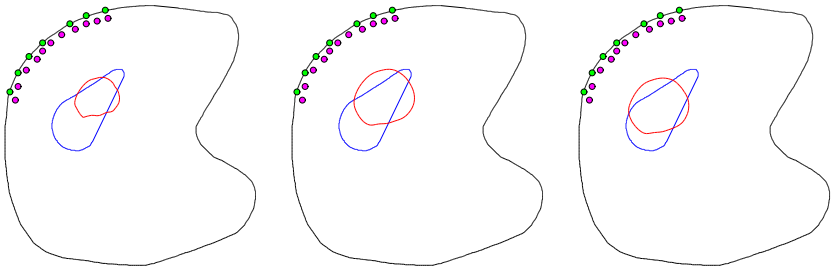
$$\frac{dE}{dt} = -\int_\Gamma (F\mathcal{N})\Gamma_t \, d\Gamma = -\int_\Gamma F\bar{F} \, d\Gamma = -\bar{F}\int_\Gamma F \, d\Gamma. \tag{197}$$

Hence, we choose

$$\bar{F} = \frac{\int_\Gamma F \, d\Gamma}{|\int_\Gamma F \, d\Gamma|} \tag{198}$$

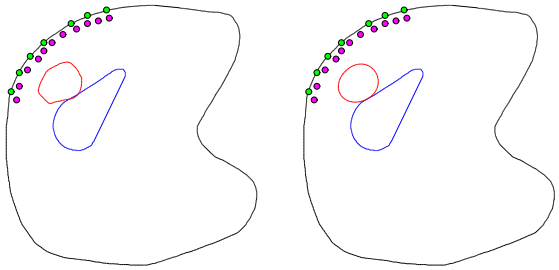
to guarantee a decrease of energy. If $\int_\Gamma F \, d\Gamma = 0$, the energy is at a minimum with regard to scaling and no further scaling should be performed.

The result of applying translation, then scaling, and then translation again is shown in Figure 67. A good recovery of location and size is obtained after only 60 iterations. However, the switch time between translation and scaling is chosen



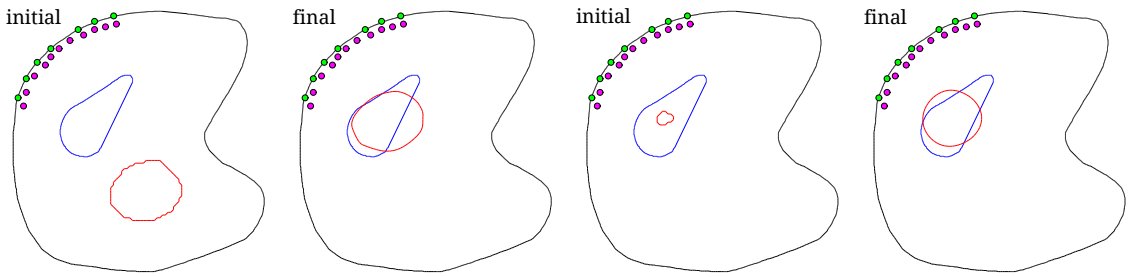
(a) iteration 37 (b) iteration 45 (c) iteration 60

Figure 67: Global search phase result for zone experiment for $\omega = 50\,\text{rad/s}$ and the joint source approach using translation (iterations 1 to 37 and 46 to 60) and scaling (iterations 38 to 45).



(a) iteration 100 (b) iteration 200

Figure 68: Global translation-scale approach running translation (iterations 1 to 100) and scaling (iterations 101 to 200) to convergence. To improve results, either size or location (center) ought to be estimated better initially as shown in Figure 69.



(a) translation with good size estimate (b) scaling with good location estimate

Figure 69: The automatic global translation-scale approach succeeds either (a) if the size estimate is good, or (b) if the location estimate is good. Shown are initial and converged contour for each case (70 iterations for translation, 18 iterations for scaling).

manually. When letting translation fully converge, one obtains Figure 68(a) and subsequent scaling does not improve the situation in Figure 68(b). The translation-scaling approach provides, therefore, an alternative to low-frequency curve evolution that has to be applied with care. Experiments with translation and scale suggest that if size is approximately correct, translation will work, and if location (center) is approximately correct, the scaling approach will work, as is illustrated in Figure 69. The combined estimation of size and location could be performed by alternating between translation and scale, or by using the full contour evolution method. Different frequencies can also affect the result, but no clear result has been observed in these experiments for how different frequencies could improve the global search.

Finally, building on the global search result from Figure 67, we apply mid-level and high frequencies manually as presented before to obtain the final reconstruction in Figure 70, which confirms that the adjoint active contour method is able to recover shape fairly accurately even if sources and receivers are restricted to a narrow zone along the outer boundary. In general, experimentation with this setup suggests that the curve segments east and south are harder to recover than segments north and west, which is attributed to the lack of illumination phenomenon.

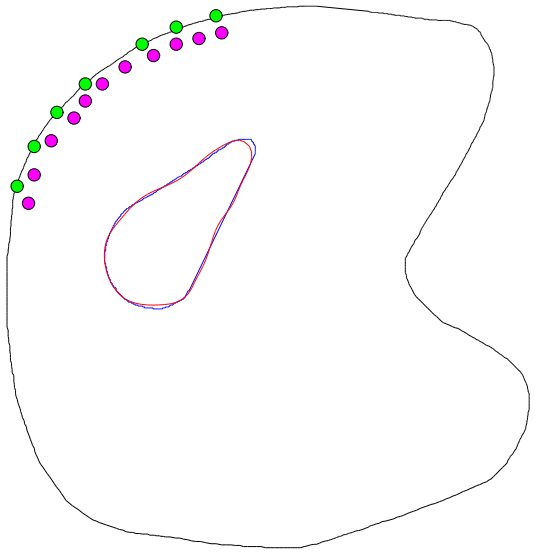


Figure 70: Final result for zone experiment using manual translation-scale, multiple frequency and multiple source approaches.

4.5.2 Elastodynamics

Elastodynamic inversion problems prove to be considerably more challenging than acoustics. First of all, computation time is increased because the problem dimension is doubled (due to the fact that field quantities are vectors as opposed to scalars) leading to the doubling of the number of rows and columns in the system matrix of the MFS. Interactive inversion becomes, therefore, more cumbersome. Moreover, the increased complexity of elastodynamic wave propagation leads to less well-behaved energies as will be demonstrated in the following sections.

We first present results for an ideal, symmetric, circular setup in Section 4.5.2.1, for which successful inversion is obtained with relative ease compared to the hand-drawn setup investigated in Section 4.5.2.2. Careful analysis of the difficulties encountered for hand-drawn (non-ideal) setups leads to the proposition of follow-up research addressing the issues, since the proposed modifications are not trivial to achieve within the framework of this thesis.

4.5.2.1 Experiment for Circular Shape

In this section, a successful inversion result is shown for the case of circular shapes and symmetric positioning of sources, receivers and collocation points. The synthetic true setup is shown in Figure 71. Arrows indicate directions of the source displacement. We consider the outer material to have the properties of acrylic (PMMA) with approximate material properties being $c_L^0 = 1915\,\mathrm{m/s}$ for the longitudinal wave speed, $c_T^0 = 856\,\mathrm{m/s}$ for the transversal (shear) wave speed and $\rho^0 = 1191\,\mathrm{kg/m^3}$ for the density. The inclusion material is assumed to be aluminum with material properties $c_L^1 = 6320\,\mathrm{m/s}$, $c_T^1 = 3130\,\mathrm{m/s}$, and $\rho^1 = 2654\,\mathrm{kg/m^3}$. The level set grid has 200 by 200 pixels and represents 4 meters in each direction. The boundary and true interface curves are discretized by 200 collocation points each, and one third of the collocation points are assigned poles for the MFS, i.e., the least-squares MFS with $P = N/3$ is applied. The pole distances are fixed to 0.25 m from the boundaries.

For a better understanding of the problem, we start with an experiment that produces an energy landscape when the curve is restricted to expanding circles. In particular, centered circles with radii ranging from 0.3 m to 0.9 m (the true circle has a radius of 0.6 m) are considered as illustrated in Figure 72. For each such circle the

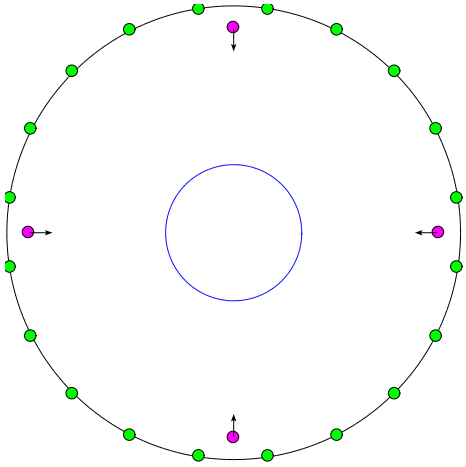


Figure 71: True setup of circular shape experiment for elastodynamics.

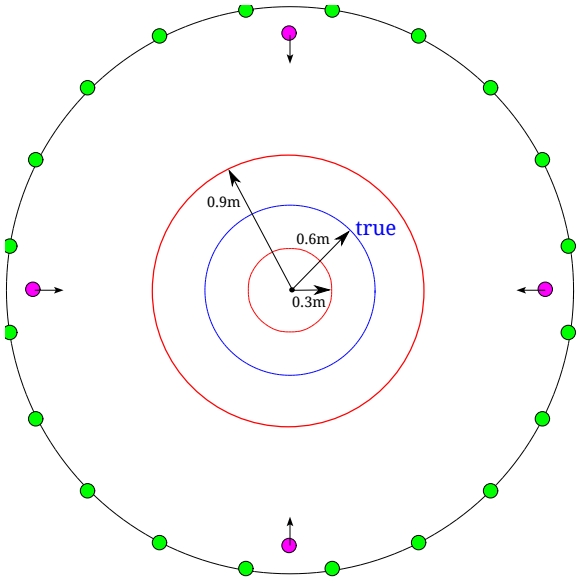


Figure 72: Setup for energy landscape using centered circular shapes. The radius of the red circle is varied between 0.3 m and 0.9 m. In Figure 73, the resulting energy is shown.

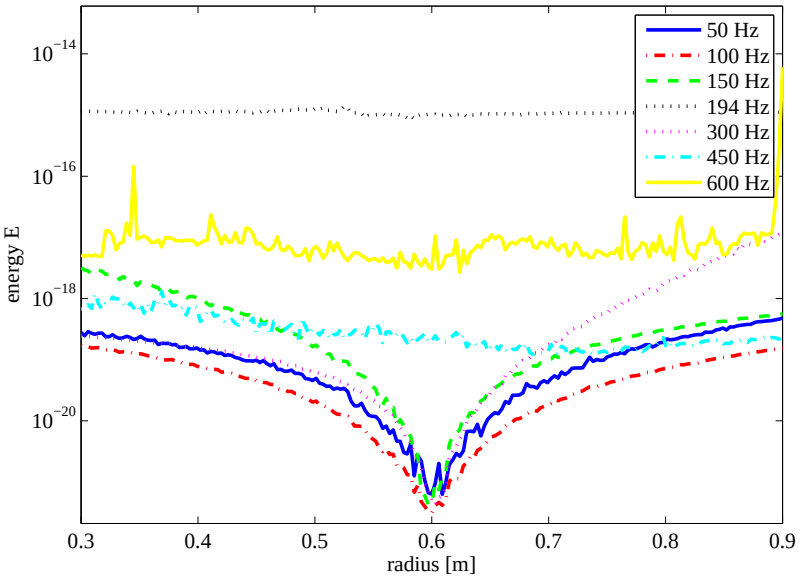


Figure 73: Energy plot over circle radius for different frequencies. For most frequencies, the true radius of 0.6 m is the global minimum, as expected.

associated energy is computed. In this way, we obtain an energy plot over the radius of the circles as is shown in Figure 73 for different frequencies.

Reassuringly, It can be observed that the minimum energy is obtained at the true

radius of 0.6 m for most frequencies. Note, however, that the energy does not drop to zero because the true and the test circle are represented differently: The true shape is represented by 200 equally spaced nodes, whereas the test circle is represented by a level set function that is initialized with pixel-accurate precision followed by 5 iterations of smoothing curvature flow. The collocation points extracted from the level set are not the same in number, not evenly distributed, and not as precisely located on the circle's circumference as the true circle's nodes.

It is also noteworthy that some frequencies are not sensitive to this experiment, particularly the higher ones. Rather surprisingly, mid-level frequencies around 194 Hz result in an almost constant energy landscape indicating that they are not useful for inversion. Frequencies around 300 Hz, however, result in a fairly smooth energy landscape with a clear minimum at $r = 0.6\,\mathrm{m}$. Low frequencies exhibit the same minimum, but are less smooth. For the computation of shape derivatives, a smooth energy landscape is desirable since this property will result in smoother gradient descent and observable decrease of the energy.

The jaggedness of some of the plots is suspected to result from a high sensitivity of the MFS's accuracy to changes in the setup. Therefore, since 300 Hz has the smoothest energy landscape, the same experiment is performed for 300 Hz, now varying the pole distances, which has been shown to be an important parameter for the MFS. The resulting energy plots in Figure 74 confirm that pole distance has a big influence, especially for setups close to the true radius, and motivate the choice of 0.25 m and 0.1 m for the pole distances on the outer surface and inner interface, respectively.

In Figure 75, successful and accurate inversion from a hand-drawn initial shape is shown mainly using 300 Hz. The use of only 300 Hz does not prove viable so that manual frequency hopping using frequencies 300 Hz, 100 Hz and 50 Hz is employed to obtain the result.

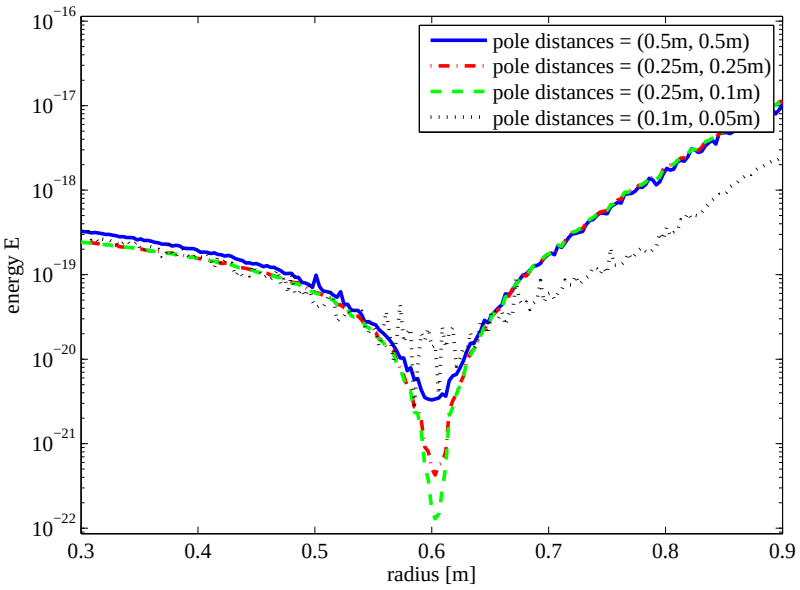
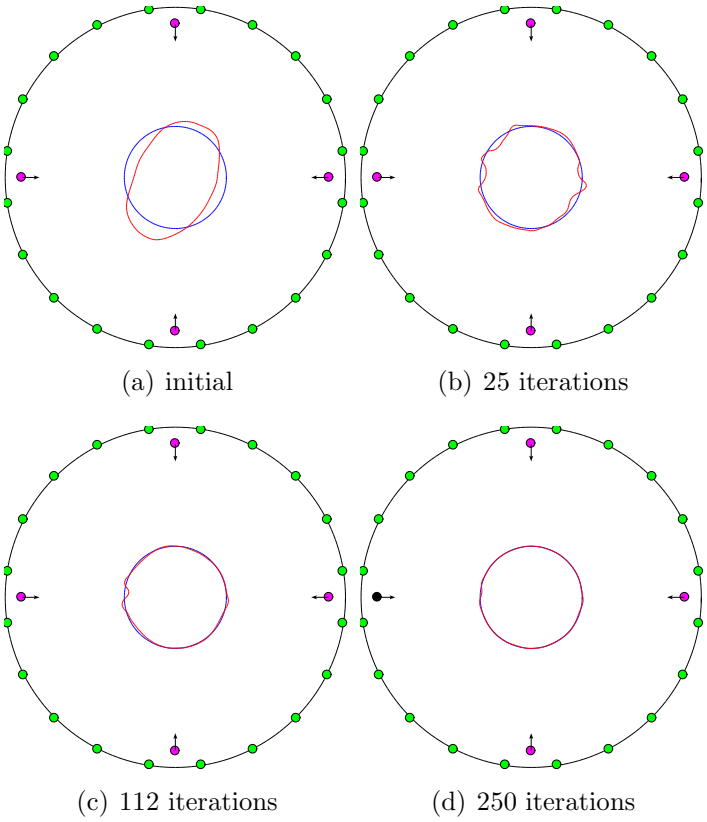


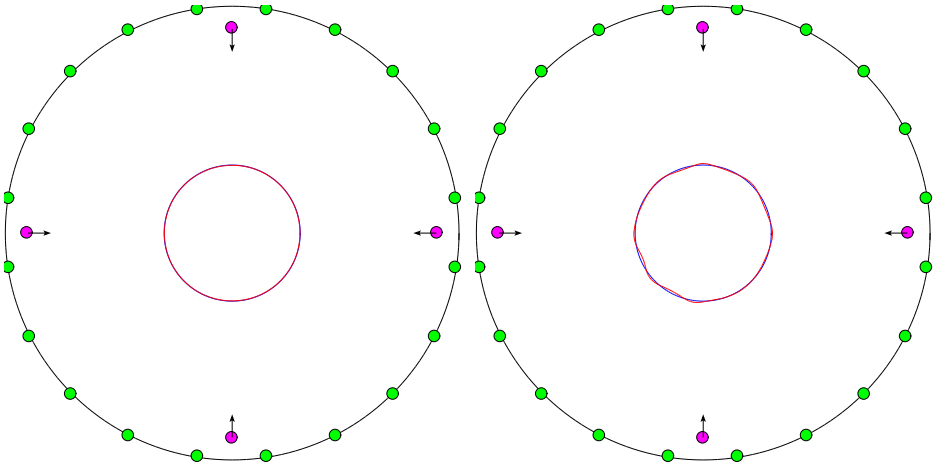
Figure 74: Energy plot over circle radius for $f = 300\,\mathrm{Hz}$ and different combinations of pole distance for $\bar{\Gamma}$ and Γ^1. The choice of pole distance is important as it affects the accuracy and smoothness of the energy landscape plot.



(a) initial (b) 25 iterations

(c) 112 iterations (d) 250 iterations

Figure 75: Successful inversion using frequency hopping at 300 Hz, 100 Hz and 50 Hz, plus joint source and source hopping approach.



(a) initial (b) 4450 iterations

Figure 76: The contour stays in the basin of attraction of the global minimum for a frequency of 300 Hz.

Finally, with good initialization and using 300 Hz, Figure 76 confirms that the contour stays attracted to the minimum with very small deviations from the initialized curve. This behavior is not for granted, as the contour has been observed to diverge for many other frequencies even with very accurate initialization. The choice of 300 Hz compares, therefore, favorably both in getting and staying attracted to the true circle.

In conclusion, this experiment shows how a good choice of frequency and MFS parameters leads to the successful inversion of a circular shape. In the following experiment, the same approach is attempted for a more general, hand-drawn shape.

4.5.2.2 Experiment for Hand-drawn Shape

The setup in Figure 77 is considered in this section and all parameters from the experiment in Section 4.5.2.1 are maintained. Both the inclusion and the surface boundary are hand-drawn shapes, and the initial guess is far away from the true shape.

A low frequency of $f = 50\,\mathrm{Hz}$ together with the translation-scale approach is used for the global search shown in Figure 78. The fast additive source approach for the

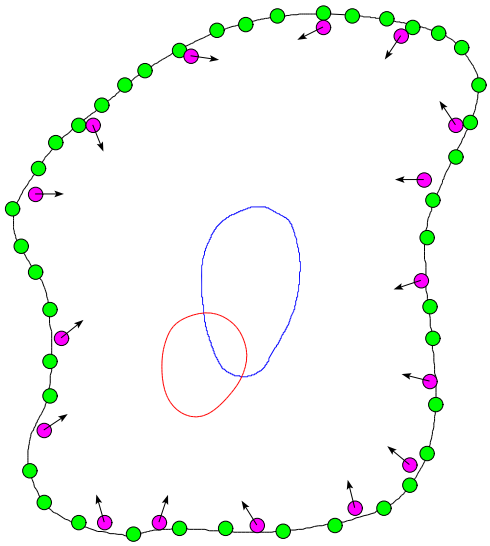
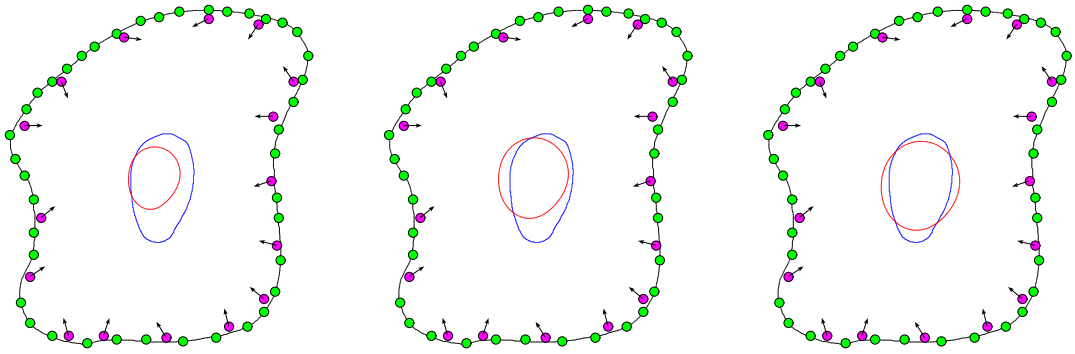


Figure 77: Hand-drawn setup for elastodynamics with true (blue) and initial (red) curve.

global search proves less successful in the elastodynamics case compared to acoustics, for which reason the more accurate joint source method is employed. This estimation of the location and size of the object provides a good initialization for the potential further recovery of shape details.

When applying manual frequency hopping as in the previous results for acoustics and elastodynamics, accurate reconstruction of shape details proves very difficult in

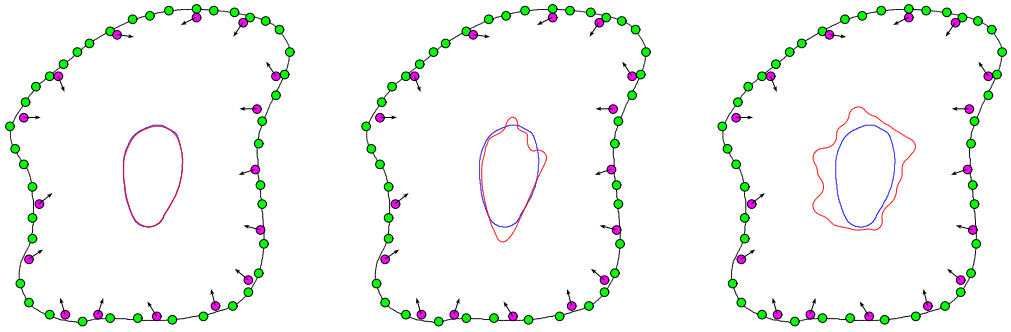


(a) 70 iterations (b) 80 iterations (c) 100 iterations

Figure 78: Alternate translation-scale global search for elastodynamics. Switch times between translation and scale are chosen at iterations 70, 80, 90 and 95.

this scenario. In fact, even very accurate initialization leads to a divergent curve flow as shown in Figure 79 for two frequencies (90 Hz and 200 Hz). However, the corresponding energy plots in Figure 80 indicate that energy is decreased up to some erratic jumps in the energy (an explanation for which will be given in the following). The adjoint active contour seems, therefore, to decrease energy as expected, but the resulting shapes are not attracted to the true shape.

To understand this issue, an experiment similar to the expanding circles in Section



(a) initialization (b) 200 iterations, $f = 90\,\text{Hz}$ (c) 200 iterations, $f = 200\,\text{Hz}$

Figure 79: Even after accurate initialization, the contour diverges (shown as an example for $f = 90\,\text{Hz}$ and $f = 200\,\text{Hz}$ after 200 iterations). The energy is minimized in this process, however, as seen in Figure 80.

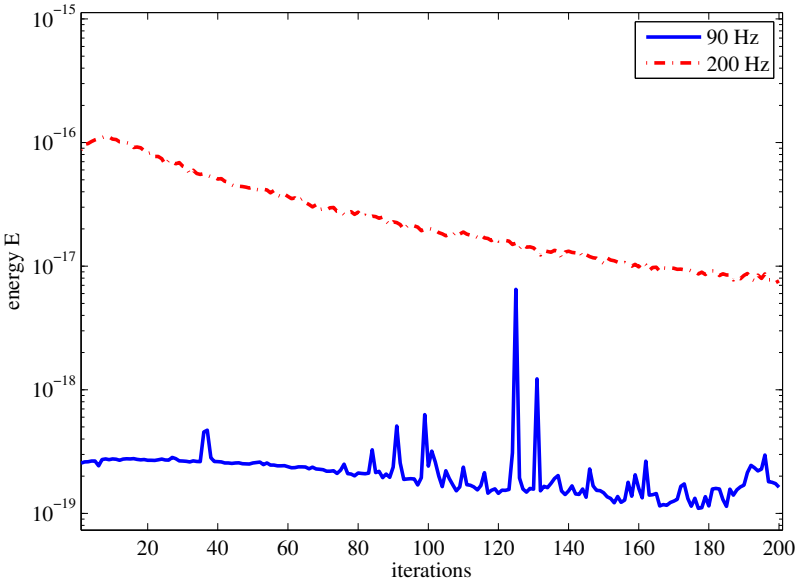


Figure 80: Energy plot over iterations for the setup in Figure 79.

4.5.2.1 is performed, where a shape is initialized very close to the true shape and then dilated ($\dot{\Gamma}_i = \mathcal{N}_i$) or eroded ($\dot{\Gamma}_i = -\mathcal{N}_i$) for a certain number of iterations. Since erosion can be viewed as "negative" dilation, we refer to both processes as dilation and indicate erosion by negative dilation iterations. In Figure 81, the maximally eroded (-50 dilation iterations) and maximally dilated (50 dilation iterations) curves are shown. The number of dilation iterations conceptually corresponds to the radius in the experiment in Section 4.5.2.1.

The thus obtained energy landscape is plotted in Figure 82 for different frequencies. This figure provides strong indications as to where the difficulties for inversion stem from. There are two main observations: First, zero dilation iterations represent a not very distinct global minimum (i.e., the energy plot is fairly flat around this point) for small frequencies and not a global minimum for higher frequencies, even though it is a local minimum for 200 Hz. Second, the energy plots are generally non-smooth.

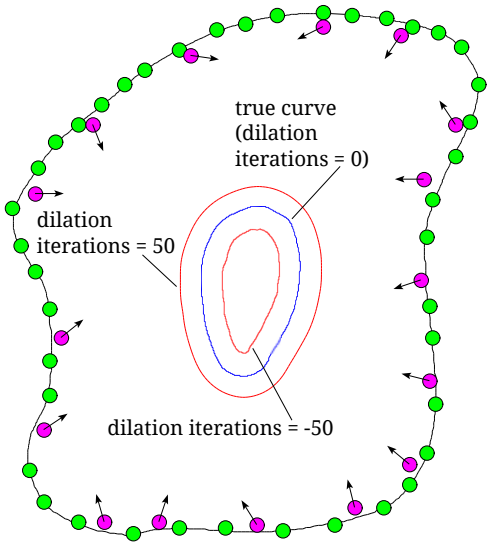


Figure 81: Setup of dilation experiment. The curve is dilated and eroded (negative dilation) from the true curve to obtain an energy plot over dilation iterations. The number of dilation iterations corresponds to the radius in the experiment with circular shapes in Section 4.5.2.1.

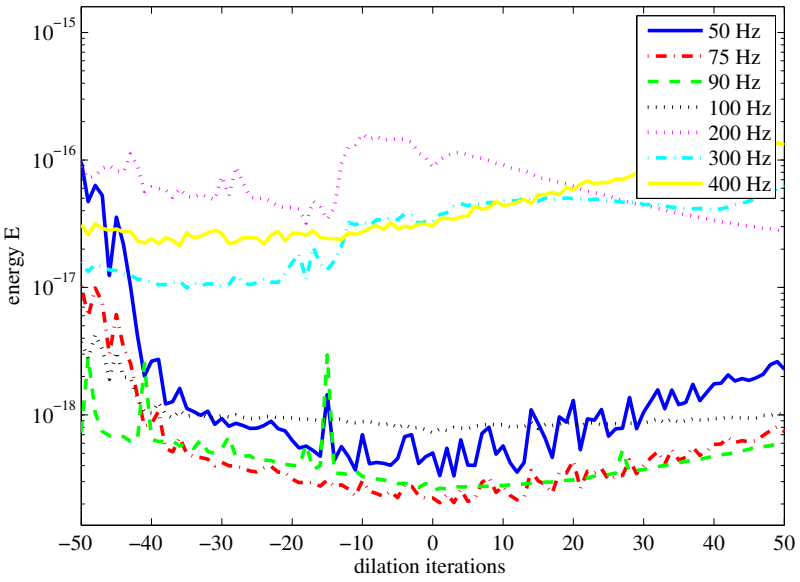


Figure 82: Energy plot over dilation iterations for different frequencies. Zero dilation iterations corresponds to the accurate initialization shown in Figure 79(a).

Looking at the energy plot for 50 Hz, one can see why the global search is successful: The energy is several orders higher when far away from the true shape and thus allows for a coarse approximation of scale (a similar observation is expected for a translation experiment). However, since the energy plots are flat near the true shape, big changes in shape lead to small changes in energy and energies are insensitive to shape changes for this reason. This property of "flatness" is believed to be a property of the adjoint active contour energy.

The observed non-smooth behavior of the energy plot, on the other hand, is believed to stem from numerical inaccuracies in the MFS forward solution, which can be viewed as noise on the forward wave field solution. Under these assumptions, the diverging results from Figure 80 can be explained as follows: The flatness of the energy around the minimizer would require very accurate computation of the shape gradient and careful choice of the step size for successful convergence. The observed inaccuracies in the forward solution lead to inaccurate shape gradients, which in turn lead to diverging shapes due to the flatness of the energy. Moreover, the result of the

dilation experiment in Figure 82 is computed for a one-dimensional search space (the dilation iterations), and is, therefore, less complex than the infinite-dimensional curve space, in which adjoint active contours evolve. In other words, only worse properties can be expected for curve evolution in comparison to a one-dimensional search space.

To corroborate the conjecture that the MFS is the source of numerical inaccuracy, one needs to compare the MFS to another numerical solver. Unfortunately, these solvers are not trivial to implement and manage, which leaves a thorough investigation beyond the scope of this thesis. Instead, a preliminary comparison to a meshfree BEM is shown next to motivate future research on numerical BVP solvers.

4.5.2.3 Preview: Meshfree Boundary Element Method

To address the aforementioned shortcomings of the MFS, BEMs are proposed as future research to improve the accuracy of forward solutions. In particular, meshfree BEMs are of interest in conjunction with level set methods for shape evolution. In this section, a promising result using constant elements following [34] chapter 2.14. is shown, which encourages future research on meshfree BEMs for adjoint active contours.

Following [34], but using our notation, BEMs exploit the Somigliana identity (assuming zero body forces)

$$u_i(\xi) = \int_\Gamma U^*_{ij}(x;\xi) t_j(x) d\Gamma_x - \int_\Gamma t^*_{ij}(x;\xi) u_j(x) d\Gamma_x, \quad \xi \in \Omega. \tag{199}$$

The boundary Γ is discretized by N linear line segments (elements) Γ_j, which are uniquely defined by their midpoints, element lengths and normal directions. Displacements u_i and tractions t_i are approximated to be constant on each element. By evaluating Eq. (199) at the midpoints of the elements (collocation points), a linear system of equations is obtained for the displacement or traction values (depending on which quantity is already specified via boundary conditions) on each element. For

one domain, this system of equations can be written as

$$\boldsymbol{H}\boldsymbol{u} = \boldsymbol{G}\boldsymbol{t}, \tag{200}$$

where $\boldsymbol{H}$ and $\boldsymbol{G}$ are matrices containing integrals of the traction and displacement fundamental solutions, respectively, along the elements. The vectors $\boldsymbol{u}$ and $\boldsymbol{t}$ contain the displacement and traction values, respectively, for each element. Since in our application, we consider traction boundary conditions, the vector $\boldsymbol{t}$ is known, and the linear system Eq. (200) is solved for the displacements $\boldsymbol{u}$.

The above BEM assumes a single domain. For adjoint active contours, the forward problem consists of coupled sub-problems for the ambient domain Ω^0 and the inclusion domain Ω^1. Let $\boldsymbol{H}^c_{ab}$ where $a, b, c \in \{0, 1\}$ denote the matrix of the integrals of traction fundamental solutions when the collocation points are on boundary Γ^a, the integrals are evaluated along Γ^b, and the equations are written for domain Ω^c. Note that $\Gamma^0 = \bar{\Gamma}$ in this section, i.e., Γ^0 is the surface boundary, whereas Γ^1 is the inclusion interface. Also, let $\boldsymbol{u}^c_b$ be the displacements values for domain Ω^c along boundary Γ^b. The systems for Ω^0 and Ω^1 are then written as

$$\begin{pmatrix} \boldsymbol{H}^0_{00} & \boldsymbol{H}^0_{01} & \boldsymbol{G}^0_{01} \\ \boldsymbol{H}^0_{10} & \boldsymbol{H}^0_{11} & -\boldsymbol{G}^0_{11} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^0_0 \\ \boldsymbol{u}^0_1 \\ \boldsymbol{t}^0_1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{G}^0_{00} \\ \boldsymbol{G}^0_{10} \end{pmatrix} \bar{\boldsymbol{t}} \tag{201}$$

and

$$\begin{pmatrix} \boldsymbol{H}^1_{11} & -\boldsymbol{G}^1_{11} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^1_1 \\ \boldsymbol{t}^1_1 \end{pmatrix} = \boldsymbol{0}, \tag{202}$$

where the continuity boundary conditions along Γ^1 have not been enforced yet. These continuity conditions yield $\boldsymbol{u}^1_1 = \boldsymbol{u}^0_1$ and $\boldsymbol{t}^1_1 = -\boldsymbol{t}^0_1$, where the minus sign arises from the fact that $\mathcal{N}^0 = -\mathcal{N}^1$. The coupled system of equations is, therefore, obtained as

$$\begin{pmatrix} \boldsymbol{H}^0_{00} & \boldsymbol{H}^0_{01} & \boldsymbol{G}^0_{01} \\ \boldsymbol{H}^0_{10} & \boldsymbol{H}^0_{11} & -\boldsymbol{G}^0_{11} \\ \boldsymbol{0} & \boldsymbol{H}^1_{11} & \boldsymbol{G}^1_{11} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^0_0 \\ \boldsymbol{u}^0_1 \\ \boldsymbol{t}^0_1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{G}^0_{00} \\ \boldsymbol{G}^0_{10} \\ \boldsymbol{0} \end{pmatrix} \bar{\boldsymbol{t}}. \tag{203}$$

Assuming the surface and inclusion interface are discretized by K^0 and K^1 elements, respectively, Eq. (203) is a $2(K^0 + 2K^1) \times 2(K^0 + 2K^1)$ system of equations that can be solved with standard linear system solvers. This method is meshfree because we do not enforce the elements to be connected. Unordered sets of elements (by defining midpoints, normals, and element sizes) can be easily extracted from the level set representation of a curve as mentioned before. Since the rows of Eq. (203) are written for each element independent of the other elements, ordering of the elements does not matter. We have, therefore, a meshfree BEM, which fits in the context of meshfree inverse scattering algorithms.

As a preview for future research, the dilation experiment from Section 4.5.2.2 is repeated using this meshfree BEM. Note one difference compared to the MFS implementation: Whereas for MFS, sources are considered to be fundamental solutions added to the overall solution, sources for BEM are realized as a unit traction boundary condition (as opposed to zero traction otherwise) on the element closest to the source, which slightly changes the formulation of the problem. The energy landscapes

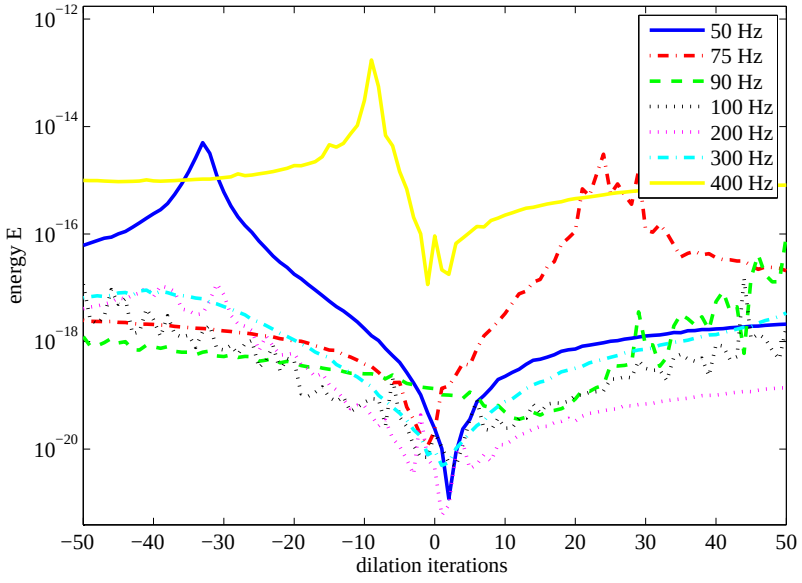


Figure 83: Energy plot over dilation iterations for different frequencies using meshfree BEM implementation, analogous to Figure 82.

using meshfree BEM are shown in Figure 83 for the same frequencies as in Figure 82. The differences between the two methods are striking. Most frequencies have a clear global minimum around 0 dilation iterations, as would be expected, and their plots are quite smooth. Especially 50 Hz and 300 Hz seem to be very promising frequencies for inversion. The fact that the minimizers are much more obvious with smoother plots is a strong indication that meshfree BEM is a more consistent and accurate method to tackle such inverse problems. Future research is proposed to focus on the further development and utilization of meshfree BEMs with extensions, for example, to linear (as opposed to constant) elements.

4.6 Conclusion

In this chapter, an algorithm has been introduced, which allows for the reconstruction of the shape of a visually hidden inclusion based on the measurement of an acoustic or elastodynamic wave process. The problem is posed as an active contour energy, which quantifies the discrepancy between measured and simulated wave amplitudes, where simulations are based on a model of wave propagation and on the geometry of the imaged domain. Energy minimization is performed using the adjoint method resulting in active contour update equations that depend on the simulated wave field and its adjoint field. Both fields are computed numerically using the MFS, a meshfree boundary value problem solver.

Synthetic results are shown for acoustics and elastodynamics. The key observation in these experiments is the dependency of the solution on a good choice of frequency. Low frequencies are capable of recovering global location and size of a shape but fail to recover finer details. Higher frequencies may recover fine shape details but diverge quickly if the current contour is far away from the true contour. The use of multiple sources has been shown to generally improve the inversion result. Three approaches of exploiting several sources have been presented and each one has its

justification in certain scenarios. For acoustics, even though generally it is possible to recover the shape with high accuracy, the success of the method depends strongly on the correct choice of parameters such as frequency and when to switch the set of parameters to avoid local minima. For elastodynamics, even though successful results have been obtained for certain regular shapes, the inversion of general shapes has proven challenging. The root cause has been identified as the insufficient accuracy of the forward solution using the MFS. Promising results using a meshfree BEM suggest that improved numerical solvers will further improve the adjoint active contour's performance in these challenging situations.

The following paths are recommended for future work on adjoint active contours:

1. Numerical forward solvers,
2. Extension to different topological configurations,
3. Extension to three dimensions,
4. Approaches for automatic parameter selection,
5. Application to real problems.

Since the experiments in this thesis indicate that the accuracy of the MFS may be insufficient, further investigations on numerical forward solvers is recommended. In Section 4.5.2.3, a meshfree BEM has been introduced with a promising result. One disadvantage of this BEM is its constant approximation along elements, which does not easily provide the gradient of the displacement field as is needed for the shape derivatives. Therefore, another meshfree BEM with linear approaximations on elements is proposed, which would allow for an easy computation of the derivative of the displacement field in the tangential direction by subtraction of the two element values. In-depth analysis and comparison between different collocation methods and well-established element methods are recommended to ensure that the numerical solver is as accurate as possible to increase the robustness of the inversion method.

Other topological configurations include, for example, half-spaces for the ambient domain. In this thesis, bounded domains are considered only. Application to seismic problems, will however, require solutions to half-space problems. Moreover, the object shapes could be layers in seismic imaging, whereas in this research, we restrict ourselves to inclusions, i.e., bounded objects. Moreover, one may consider several inclusions which affect each other in the forward solution and the inversion scheme. Note, however, that the shape gradients are already provided for several interfaces so that the extension to multiple objects is simply a matter of handling several contours and computing forward BVP solutions.

The extension to three dimensions is necessary for many real applications since only a limited number of real problems can be approximated by a two-dimensional problem. The adjoint active contour method is, in theory, easily extended to adjoint active surfaces, and in fact, the shape gradients provided in this thesis are the same for two and for three dimensions. Again, the main difficulty in moving to three dimensions comes at the implementation level where both the level set code and the forward BVP solver need to be adapted to work in three dimensions. Especially numerical BVP solvers for three dimensions require much effort since the requirements for bookkeeping and meshing for element methods are much higher. If meshfree methods turn out to provide accurate results in three dimensions, it would considerably simplify the forward problems of adjoint active surfaces by not having to mesh the boundaries or domains. In other words, the full benefit of using meshfree collocation methods can be harvested in three dimensions where meshing is considerably harder than in two dimensions.

The large amount of user input needed to obtain the results in this thesis raises the important question of how to choose parameters and switching times automatically or semi-automatically when knowledge of the true shape is not available. The straightforward answer to this question is more computation power, where one could evolve

contours using several sets of parameters in parallel and check which performs best. Moreover, accurate forward solutions become even more important since the energy value for a given shape is the only piece of information about the state of the inversion, so that errors coming from numerical inaccuracies should be minimized as much as possible.

Finally, inversion of data from controlled, real experiments is recommended. Acoustic chambers (air) or water pools (water) provide feasible test environments to acquire data modeled by the Helmholtz equation. Data described by elastodynamics can be obtained with the help of non-destructive testing labs, where one could, for example, try to locate a hole in an otherwise intact plate. Seismic experiments are of interest as well. However, they hardly represent a controlled environment so that inversion of such data might prove difficult as a trial experiment for adjoint active contours.

CHAPTER V

CONCLUSION

In this thesis, three novel applications of physics-inspired or physics-driven variational methods have been presented. The two computer vision projects deal with novel optical flow and active contour formulations for the specific tasks of flame detection in visible-spectrum videos and shape-plus-radiance-based segmentation in infrared images. With adjoint active contours, we have applied similar methods to shape-based imaging, an emerging image formation technique.

5.1 Summary of Contributions

The contributions of the first project (flame detection) are

- development of a complete detection pipeline with active-contour-based candidate region tracking and optical flow features,
- novel and application-specific formulation of optical flow estimators,
- novel region-based feature extraction from optical flow fields.

The contributions of the second project (photo-geometric features) are

- generalization of Mumford-Shah through anisotropic diffusion,
- definition of an efficient, one-dimensional radiance feature function (photo-geometric representation) of an object that couples shape and radiance of an object,
- formulation of a recognition-segmentation energy with efficient coupled shape-plus-radiance priors.

The contributions of the third project (adjoint active contours) are

- complete derivation and presentation of the meshfree inversion algorithm for two-dimensional inclusion problems in acoustics and elastodynamics,
- extensive synthetic results and in-depth analysis of variants of the method for acoustics,
- proof-of-concept synthetic results and problem analysis for elastodynamics.

5.2 Future Work

While the computer vision projects can be considered fairly completed in the sense that the algorithms are shown to work on real images and could, therefore, be considered for real-world applications, the adjoint active contours project has just left its proof-of-concept stage, where the method has been analyzed on synthetic data. For this reason, the main future work is recommended for this last project by applying the method to real measurements of problems in geophysics, medical imaging and non-destructive testing. The gap between models and reality in these fields may prove a major challenge to overcome since inverse scattering is already challenging in ideal, synthetic situations, but expectations for this method to succeed are certainly not unsubstantiated, as some of the results in this thesis suggest. More detailed suggestions for future work can be found in the concluding remarks of each project's chapter.

5.3 Concluding Remark

We conclude this thesis with some insight into the connection between inverse scattering and region-based segmentation in computer vision, which will bring the two parts of this thesis (computer vision and shape-based imaging) together in one generalized way.

Consider again the Mumford-Shah energy

$$E_{MS}(C,f) = \alpha \int_{\Omega} (I-f)^2 \, d\Omega + \beta \left\{ \int_{R} \|\nabla f\|^2 \, d\Omega + \int_{R^c} \|\nabla f\|^2 \, d\Omega \right\}, \tag{204}$$

where we ignore the curve smoothness term because it is irrelevant in the following. As mentioned earlier, this energy models radiance as a piece-wise smooth function and edges may occur at the interface curve C. In other words, if the image I is indeed piece-wise smooth with strong edges at the interface, then we may expect good segmentation results. If one of the two assumptions is violated, the algorithm's chance of failure increases. Successful segmentation with the Mumford-Shah energy depends in large on the image satisfying the underlying model.

Now, consider the inverse scattering energy

$$E(\Gamma) = \frac{1}{2} \sum_{r=1}^{R} \|u(x_r) - \hat{u}(x_r)\|^2, \tag{205}$$

where u is the simulated wave field and $\hat{u}(x_r)$ are the receiver measurements. The concept of modeling is obvious in this case, since u is *simulated* exactly according to the physical model we choose. It is obvious that we cannot expect good results if the model (and thus the simulated wave field u) deviates from the real world and what we measure from it.

From the above, we draw the following analogies between these two energies: Given some data (that is: the image I for image segmentation; and the measured waves $\hat{u}$ for inverse scattering) we build a model (that is: a piece-wise smooth radiance image f for image segmentation; and a simulated wave field u for inverse scattering) and formulate an energy that penalizes the difference between the data and the model (fidelity term) depending on the input curve. The minimum curve is expected to best "explain" the data according to the model.

Within this generalized view on two very different problems, the following difference can be observed: In image segmentation, we usually have a large amount of data, i.e., a dense set of pixel values, which is defined over the whole domain. We know pretty well what the data looks like, but we are not so sure about the location of object and background region because often times we do not have an accurate model. In Mumford-Shah, the model is actually quite crude assuming that radiance is

piece-wise smooth. The incorporation of prior knowledge such as shape and radiance precisely attempts to increase the accuracy of the model and thus the robustness of the algorithm. In other words, in image segmentation, problems of robustness usually arise from the lack of accurate models, i.e., understanding of the data.

In inverse scattering, the problem is quite the opposite: Models for wave propagation are fairly accurate and deterministic. The science of physics provides accurate models of the data. The problem, however, for certain applications, such as seismic probing, is the lack of measured data. First of all, measurements are usually available only on the surface of the domain as opposed to image segmentation, where data is available in the whole image domain. Furthermore, taking dense measurements on a surface may take much effort in practice, which is why the inverse scattering approach in this thesis considers only few measurements. Thus, the problem of ill-posedness arises from the lack of data, and meaningful recovery of the interface is possible only due to accurate physical models.

From this perspective, these two problems are at opposite poles of this "problem space" spanned by measurement density and model accuracy. Their difficulties arise from a lack of either one property, the difficulties that make them interesting for research.

APPENDIX A

DERIVATION OF ANISOTROPIC MUMFORD-SHAH

A.1 General Case

For a given level set function ψ such that $\nabla\psi \neq 0$ almost everywhere, the general form of the anisotropic Mumford-Shah energy is defined as

$$E_{\text{aMS}}(C,f) = \alpha\int_{\Omega}(I-f)^2\,d\Omega + \beta_{\perp}^{in}\int_{R}\|\nabla_{\perp}f\|^2\,d\Omega + \beta_{\parallel}^{in}\int_{R}\|\nabla_{\parallel}f\|^2\,d\Omega + ... \\ + \beta_{\perp}^{out}\int_{R^c}\|\nabla_{\perp}f\|^2\,d\Omega + \beta_{\parallel}^{out}\int_{R^c}\|\nabla_{\parallel}f\|^2\,d\Omega + \gamma\int_{C}1\,d\Omega. \tag{206}$$

where the gradient operator ∇ is orthogonally decomposed at each point into its action parallel to the level set of ψ passing through that point ($\nabla_{\parallel}$) and its action perpendicular to the same level set ($\nabla_{\perp}$). This decomposition may be obtained by left-multiplying the gradient operator ∇ by appropriate projection matrices $P_{\perp}$ and $P_{\parallel}$ which are both functions of $\nabla\psi$.

$$\nabla_{\perp} = P_{\perp}\nabla, \quad \text{where } P_{\perp} = \frac{\nabla\psi\nabla^T\psi}{\|\nabla\psi\|^2}, \tag{207a}$$

$$\nabla_{\parallel} = P_{\parallel}\nabla, \quad \text{where } P_{\parallel} = \mathcal{I} - \frac{\nabla\psi\nabla^T\psi}{\|\nabla\psi\|^2}, \tag{207b}$$

and $\mathcal{I}$ is the identity matrix.

We consider only the interior integral term since the calculus for the exterior term can be obtained analogously. For notational convenience, we denote $\beta_{\perp} := \beta_{\perp}^{in}$, $\beta_{\parallel} := \beta_{\parallel}^{in}$, and $f := f|_R$ is restricted to R. The interior part of E_{aMS} is defined as

$$E_{\text{aMS}}^{in}(C,f,\psi) = \int_{R}\alpha(I-f)^2 + \beta_{\perp}\|\nabla_{\perp}f\|^2 + \beta_{\parallel}\|\nabla_{\parallel}f\|^2 dx. \tag{208}$$

A.1.1 Euler-Lagrange PDE for Radiance Function

For a fixed choice of C and ψ, the Euler-Lagrange equations provide the following partial differential equation (PDE) for the optimal radiance function f^*

$$-\alpha(I - f^*) - \beta_\perp \nabla \cdot \nabla_\perp f^* - \beta_\parallel \nabla \cdot \nabla_\parallel f^* = 0 \tag{209}$$

subject to the boundary condition

$$\left(\beta_\perp \nabla_\perp f^* + \beta_\parallel \nabla_\parallel f^*\right) \cdot \mathcal{N} = 0. \tag{210}$$

To carry out the necessary derivatives of the Euler-Lagrange formula, the local gradient components are written in matrix form according to Eq. (207), e.g.,

$$\|\nabla_\perp f\|^2 = (\nabla f)^T P_\perp^T P_\perp (\nabla f), \tag{211}$$

and simplified by making use of the facts that $P_\perp = P_\perp^T$ and $P_\parallel = P_\parallel^T$, as well as $P_\perp P_\perp = P_\perp$ and $P_\parallel P_\parallel = P_\parallel$.

Notice that in the case, where the boundary curve C is itself a level set of ψ, $\nabla_\parallel f^* \cdot \mathcal{N} = 0$ and so $\nabla_\perp f^* \cdot \mathcal{N} = \nabla f^* \cdot \mathcal{N}$ for any differentiable f^* along the boundary C, thereby reducing the above condition to the standard Neumann boundary condition $\nabla f^* \cdot \mathcal{N} = 0$.

A.1.2 Curve Gradient for Arbitrary Level Set Function

Introducing artificial time t and letting the unknowns C, ψ, and f depend on t, the total time derivative can be written as

$$\frac{dE^{in}_{\text{aMS}}}{dt} = \left\langle \nabla_C E^{in}_{\text{aMS}}, C_t \right\rangle + \left\langle \frac{\partial E^{in}_{\text{aMS}}}{\partial f}, f_t \right\rangle + \left\langle \frac{\partial E^{in}_{\text{aMS}}}{\partial \psi}, \psi_t \right\rangle. \tag{212}$$

For now, it is assumed that ψ is an arbitrary level set function independent of C. For fixed f and ψ, the time derivative of the energy becomes

$$\frac{dE^{in}_{\text{aMS}}(C)}{dt} = \int_C \left(\alpha(I - f)^2 + \beta_\perp \|\nabla_\perp f\|^2 + \beta_\parallel \|\nabla_\parallel f\|^2\right) \frac{\partial C}{\partial t} \cdot \mathcal{N}\, ds \tag{213}$$

and so

$$\nabla_C E^{in}_{\text{aMS}} = \left(\alpha(I-f)^2 + \beta_\perp \|\nabla_\perp f\|^2 + \beta_\parallel \|\nabla_\parallel f\|^2\right)\mathcal{N}. \tag{214}$$

For fixed C and ψ, the time derivative of the energy becomes

$$\begin{aligned}\frac{dE^{in}_{\text{aMS}}(f)}{dt} &= 2\int_R -\alpha(I-f)\frac{\partial f}{\partial t} + \beta_\perp \nabla_\perp f \cdot \nabla \frac{\partial f}{\partial t} + \beta_\parallel \nabla_\parallel f \cdot \nabla \frac{\partial f}{\partial t}\, dx \\ &= -2\int_R \left(\alpha(I-f) + \beta_\perp \nabla\cdot\nabla_\perp f + \beta_\parallel \nabla\cdot\nabla_\parallel f\right)\frac{\partial f}{\partial t}\, dx + ... \\ &\quad + 2\int_C \left(\beta_\perp \nabla_\perp f + \beta_\parallel \nabla_\parallel f\right)\cdot \mathcal{N}\frac{\partial f}{\partial t}\, ds, \end{aligned} \tag{215}$$

after applying Green's theorem. When substituting in the optimal radiance f^*, Eq. (210) and Eq. (211) make the above terms vanish, meaning that $\frac{\partial E^{in}_{\text{aMS}}}{\partial f}(C, f^*, \psi) = 0$. This result was expected since f^* satisfies the first-order necessary condition of the minimization problem. For the last term of Eq. (212), we assume that C and f be fixed, and ψ (and therefore $P_\perp$ and $P_\parallel$) depends on t. Consider the following preliminary calculations

$$\begin{aligned}\frac{\partial P_\perp}{\partial t} &= \frac{\partial}{\partial t}\left(\frac{\nabla\psi\nabla^T\psi}{\|\nabla\psi\|^2}\right) \\ &= \left(\mathcal{I} - \frac{\nabla\psi\nabla^T\psi}{\|\nabla\psi\|^2}\right)\frac{\nabla\frac{\partial\psi}{\partial t}\nabla^T\psi}{\|\nabla\psi\|^2} + \frac{\nabla\psi\nabla^T\frac{\partial\psi}{\partial t}}{\|\nabla\psi\|^2}\left(\mathcal{I} - \frac{\nabla\psi\nabla^T\psi}{\|\nabla\psi\|^2}\right) \\ &= P_\parallel \frac{\nabla\frac{\partial\psi}{\partial t}\nabla^T\psi}{\|\nabla\psi\|^2} + \frac{\nabla\psi\nabla^T\frac{\partial\psi}{\partial t}}{\|\nabla\psi\|^2}P_\parallel \end{aligned} \tag{216a}$$

$$\frac{\partial P_\parallel}{\partial t} = \frac{\partial}{\partial t}(\mathcal{I} - P_\perp) = -\frac{\partial P_\perp}{\partial t} \tag{216b}$$

which leads to these expressions for the derivatives of $\|\nabla_\perp f\|^2$ and $\left\|\nabla_\parallel f\right\|^2$

$$\begin{aligned}\frac{\partial}{\partial t}\|\nabla_\perp f\|^2 &= \frac{\partial}{\partial t}\left(\nabla^T f P_\perp \nabla f\right) \;=\; \nabla^T f \frac{\partial P_\perp}{\partial t}\nabla f \\ &= \nabla^T f\left(P_\parallel \frac{\nabla\frac{\partial\psi}{\partial t}\nabla^T\psi}{\|\nabla\psi\|^2} + \frac{\nabla\psi\nabla^T\frac{\partial\psi}{\partial t}}{\|\nabla\psi\|^2}P_\parallel\right)\nabla f, \\ &= 2\nabla_\parallel^T f \frac{\nabla\frac{\partial\psi}{\partial t}\nabla^T\psi}{\|\nabla\psi\|^2}\nabla f \end{aligned} \tag{217a}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left\|\nabla_\parallel f\right\|^2 &= \frac{\partial}{\partial t}\left(\nabla^T f P_\parallel \nabla f\right) \;=\; \nabla^T f \frac{\partial P_\parallel}{\partial t}\nabla f \\ &= -\nabla^T f \frac{\partial P_\perp}{\partial t}\nabla f \;=\; -\frac{\partial}{\partial t}\|P_\perp \nabla f\|^2. \end{aligned} \tag{217b}$$

Using these equations, the partial energy gradient for ψ becomes

$$\begin{aligned}
\frac{dE^{in}_{\text{aMS}}(\psi)}{dt} &= \int_R \frac{\partial}{\partial t}\left(\beta_\perp\|\nabla_\perp f\|^2 + \beta_\parallel\|\nabla_\parallel f\|^2\right)\,dx \\
&= \int_R (\beta_\perp - \beta_\parallel)\frac{\partial}{\partial t}\|\nabla_\perp f\|^2\,dx \\
&= 2\int_R (\beta_\perp - \beta_\parallel)\nabla_\parallel^T f \frac{\nabla\frac{\partial\psi}{\partial t}\nabla^T\psi}{\|\nabla\psi\|^2}\nabla f\,dx \\
&= 2\int_R (\beta_\perp - \beta_\parallel)\nabla\frac{\partial\psi}{\partial t}\cdot\left(\frac{\partial f}{\partial\eta}\frac{\nabla_\parallel f}{\|\nabla\psi\|}\right)\,dx \\
&= -2(\beta_\perp - \beta_\parallel)\left(\int_R \frac{\partial\psi}{\partial t}\nabla\cdot\left(\frac{\frac{\partial f}{\partial\eta}}{\|\nabla\psi\|}\nabla_\parallel f\right)dx - \int_C \frac{\partial\psi}{\partial t}\frac{\frac{\partial f}{\partial\eta}}{\|\nabla\psi\|}\nabla_\parallel f\cdot\mathcal{N}\,ds\right) \quad (218)
\end{aligned}$$

after applying Green's theorem in the last line. Notice that we have denoted the directional derivative of f along unit gradient direction $\frac{\nabla\psi}{\|\nabla\psi\|}$ of ψ by $\frac{\partial f}{\partial\eta}$, representing, in effect, the scalar part of the vector-valued orthogonal gradient $\nabla_\perp f$. More precisely, it is defined as $\frac{\partial f}{\partial\eta} = \nabla f\cdot\frac{\nabla\psi}{\|\nabla\psi\|}$, allowing us to express the orthogonal gradient as $\nabla_\perp f = \frac{\partial f}{\partial\eta}\frac{\nabla\psi}{\|\nabla\psi\|}$.

Now, combining all these results and plugging in the optimal radiance function f^*, we obtain the total gradient (assuming that ψ is an arbitrary level set function) as

$$\begin{aligned}
\frac{dE^{in}_{\text{aMS}}(C,f^*,\psi)}{dt} &= \int_C \left(\alpha(I-f^*)^2 + \beta_\perp\|\nabla_\perp f^*\|^2 + \beta_\parallel\|\nabla_\parallel f^*\|^2\right)\frac{\partial C}{\partial t}\cdot\mathcal{N}\,ds + ... \\
&-2(\beta_\perp - \beta_\parallel)\left[\int_R \frac{\partial\psi}{\partial t}\nabla\cdot\left(\frac{\frac{\partial f^*}{\partial\eta}}{\|\nabla\psi\|}\nabla_\parallel f^*\right)dx - \int_C \frac{\partial\psi}{\partial t}\frac{\frac{\partial f^*}{\partial\eta}}{\|\nabla\psi\|}\nabla_\parallel f^*\cdot\mathcal{N}\,ds\right].
\end{aligned} \quad (219)$$

A.1.3 Relationship between Curve and Level Set Function

Now we connect ψ and C and their respective perturbations in the parameter t by assuming that ψ is a L^2-signed-distance function for C, which means that C itself is the zero level set of ψ and that all other level sets of ψ are uniquely determined by C as well. We will therefore adopt the notation ψ_C to reflect this coupling, where ψ_C is

defined as

$$\psi_C(x) := \pm \min_p \|x - C(p)\| = \pm\|x - C(p^*)\| \tag{220}$$

$$\text{where} \quad p^*(x) := \arg\min \|x - C(p)\|, \tag{221}$$

where the convention $\psi_C(x) > 0$ for $x \notin R$ and $\psi_C(x) < 0$ for $x \in R$ is used, and $\|\cdot\|$ denotes the L^2-norm. The variable p parameterizes the boundary C (zero level set of ψ_C) and p^* denotes a specific parameter value associated to a boundary point which achieves the minimum distance ψ to the given interior point x. Note that p^* is only unique "almost everywhere" since it will have multiple values at points along the symmetry set. For additional convenience we introduce the notation

$$C^*(x) := C(p^*) \quad \text{and} \quad \mathcal{N}^*(x) := \mathcal{N}(p^*) \tag{222}$$

to denote the closest boundary point C^* as well as the outward unit normal $\mathcal{N}^*$ at that closest point. Notice that by this notation convention, symbols with the asterisk are all functions of the spatial variable x, whereas the matching symbols without the asterisk superscript are functions of the boundary parameter p.

Assuming $C(p)$ is differentiable, then a necessary condition for p^*, whether it is unique or not, is

$$0 = \left.\frac{\partial}{\partial p}\right|_{p=p^*} \|x - C(p)\| = \frac{x - C^*}{\|x - C^*\|} \cdot \left.\frac{\partial C}{\partial p}\right|_{p=p^*}, \tag{223}$$

which means that

$$\frac{x - C^*}{\|x - C^*\|} = \pm\mathcal{N}^* \tag{224}$$

or equivalently

$$x - C^* = \psi_C \mathcal{N}^*. \tag{225}$$

This says that the line from the closest point $C^*(x)$ to x itself is aligned with the unit normal $\mathcal{N}^*$ at that closest point. As such, we may interpret this line segment as

a "radial line" between x and the boundary. Applying the gradient operator to this condition and noting that $\mathcal{N}^*$ is orthogonal to the tangent vector $\frac{\partial C}{\partial p}$, we obtain

$$0 = \nabla\left((x - C^*) \cdot \frac{\partial C}{\partial p}\right) = \left(\mathcal{I} - \nabla p^* \frac{\partial C}{\partial p}^T\right) \frac{\partial C}{\partial p} + \nabla p^* \frac{\partial^2 C}{\partial p^2}^T (x - C^*), \tag{226}$$

or equivalently,

$$\frac{\partial C}{\partial p} = \nabla p^* \left(\left\|\frac{\partial C}{\partial p}\right\|^2 - \frac{\partial^2 C}{\partial p^2} \cdot (x - C^*)\right) = \nabla p^* \left(\left\|\frac{\partial C}{\partial p}\right\|^2 - \psi_c \frac{\partial^2 C}{\partial p^2} \cdot \mathcal{N}^*\right). \tag{227}$$

Then, taking the dot product with $\mathcal{N}^*$ on both sides reveals the generic condition

$$\nabla p^* \cdot \mathcal{N}^* = 0, \tag{228}$$

which means that p^* (when it is unique) does not change as we move along the normal direction $\mathcal{N}^*$ to the closest boundary point C^*, and therefore C^* remains the closest point. Continuing the calculation by substituting

$$-\frac{\partial^2 C}{\partial p^2} \cdot \mathcal{N}^* = \kappa^* \left\|\frac{\partial C}{\partial p}\right\|^2 \tag{229}$$

in Eq. (227) yields

$$\frac{\partial C}{\partial p} = \nabla p^* \left(1 + \kappa^* \psi_C\right) \left\|\frac{\partial C}{\partial p}\right\|^2 \tag{230}$$

resulting in the following expression for ∇p^*

$$\nabla p^* = \frac{\frac{\partial C}{\partial p}}{\left(1 + \kappa^* \psi_C\right) \left\|\frac{\partial C}{\partial p}\right\|^2}. \tag{231}$$

We may now compute the derivatives of the signed distance function ψ_C starting with the spatial gradient

$$\begin{aligned}
\nabla \psi_C &= \pm \frac{\left(\mathcal{I} - \nabla p^* \left(\frac{\partial C}{\partial p}\right)^T\right)(x - C^*)}{\|x - C^*\|} = \frac{\left(\mathcal{I} - \frac{\partial C}{\partial p} \nabla^T p^*\right)(\psi_C \mathcal{N}^*)}{\pm \|x - C^*\|} \\
&= \frac{\left(\mathcal{I} - \frac{\partial C}{\partial p} \nabla^T p^*\right)(\psi_C \mathcal{N}^*)}{\psi_C} = \left(\mathcal{I} - \frac{\partial C}{\partial p} \nabla^T p^*\right) \mathcal{N}^* \\
&= \mathcal{N}^* - \frac{\partial C}{\partial p} (\underbrace{\nabla p^* \cdot \mathcal{N}^*}_{0}) = \mathcal{N}^*,
\end{aligned} \tag{232}$$

which gives the well-known Eikonal equation

$$\|\nabla\psi_C\| = 1. \tag{233}$$

If we now consider all of these functions (ψ_C, C, p^*, C^*, $\mathcal{N}^*$) to vary with the artificial time parameter as well, we obtain

$$\frac{\partial\psi_C}{\partial t} = \pm\frac{\frac{\partial(x-C^*)}{\partial t}\cdot(x-C^*)}{\|x-C^*\|} = \frac{-\frac{\partial(C^*)}{\partial t}\cdot(\psi_C\mathcal{N}^*)}{\psi_C} = -\frac{\partial C^*}{\partial t}\cdot\mathcal{N}^*. \tag{234}$$

Since this expression only depends upon the normal component of $\frac{\partial C^*}{\partial t}$, we may exploit the following convenient relationship,

$$\frac{\partial C^*}{\partial t}\cdot\mathcal{N}^* = \left(\frac{\partial C}{\partial t} + \frac{\partial C}{\partial p}\frac{\partial p^*}{\partial t}\right)\Bigg|_{p=p^*}\cdot\mathcal{N}^* = \left.\frac{\partial C}{\partial t}\right|_{p=p^*}\cdot\mathcal{N}^* \tag{235}$$

by Eq. (228) and Eq. (231). This relationship shows that the normal component of the evolution $\frac{\partial C^*}{\partial t}$ of the closest point C^* on the evolving boundary C exactly matches the normal component of the evolution $\frac{\partial C}{\partial t}$ of the boundary itself, i.e., $\frac{\partial C}{\partial t}$ and $\frac{\partial C^*}{\partial t}$ can only differ by a tangential component. This property greatly simplifies matters as it allows us to express the evolution of the distance function $\frac{\partial\psi_c}{\partial t}$ directly from the evolution $\frac{\partial C}{\partial t}$ of the boundary without having to account for the more complicated evolution $\frac{\partial C^*}{\partial t}$ of the closest point.

Finally, we note that in the neighborhood of a point x for which $C^*(x)$ is unique, we may utilize curvilinear coordinates p and τ, where p, as before, parametrizes the boundary C plus all the other level sets, and where τ represents the distance to the boundary, i.e., by definition, the level set value ψ_C. Thereby, τ acts as the radial coordinate from the boundary by our earlier interpretation. The mapping from these curvilinear coordinates to x is obtained by inverting Eq. (225) to obtain

$$x(p,\tau) = C(p) + \tau\mathcal{N}(p) \tag{236}$$

with Jacobian and determinant given by

$$\det\begin{bmatrix}\frac{\partial x}{\partial p} & \frac{\partial x}{\partial \tau}\end{bmatrix} = \det\begin{bmatrix}\left(\frac{\partial C}{\partial p} + \tau\frac{\partial N}{\partial p}\right) & N\end{bmatrix} = \det\begin{bmatrix}\left(\frac{\partial C}{\partial s} + \tau\frac{\partial N}{\partial s}\right)\|\frac{\partial C}{\partial p}\| & N\end{bmatrix}$$

$$= \det\left[(1+\tau\kappa)\,T\left\|\tfrac{\partial C}{\partial p}\right\| \quad N\right]$$
$$= (1+\tau\kappa)\left\|\frac{\partial C}{\partial p}\right\| \det\begin{bmatrix} T & N \end{bmatrix} = (1+\tau\kappa)\left\|\frac{\partial C}{\partial p}\right\|, \tag{237}$$

thereby yielding the following change of variable relationship in the corresponding integral measures

$$dx = (1+\tau\kappa)\left\|\frac{\partial C}{\partial p}\right\| dp\, d\tau = (1+\tau\kappa) ds\, d\tau, \tag{238}$$

where $ds := \left\|\frac{\partial C}{\partial p}\right\| dp$. With these coordinates, the gradient operator can be decomposed as follows

$$\nabla = (\nabla p^*)\frac{\partial}{\partial p} + (\nabla\tau)\frac{\partial}{\partial \tau} = \left(\frac{\frac{\partial C}{\partial p}}{(1+\kappa^*\tau)\left\|\frac{\partial C}{\partial p}\right\|^2}\right)\frac{\partial}{\partial p} + \mathcal{N}^*\frac{\partial}{\partial \tau}$$
$$= \frac{\partial C^*}{\partial s}\left(\frac{\frac{\partial}{\partial s}}{1+\kappa^*\tau}\right) + \mathcal{N}^*\frac{\partial}{\partial \tau}$$
$$= \frac{\partial C^*}{\partial s}\frac{\partial}{\partial s_\tau} + \mathcal{N}^*\frac{\partial}{\partial \tau} = \nabla_\parallel + \nabla_\perp, \tag{239}$$

where ds_τ denotes the arclength element of the τ-level set C_τ, which is related to the arclength element ds of the zero level set (of the boundary C) according to

$$\frac{\partial s_\tau}{\partial s} = 1 + \kappa^*\tau. \tag{240}$$

In these coordinates, integral over the region R can be expressed as an outer integral along the boundary C in its arclength parameter s with an inner integral along each radial line in the distance parameter τ. Note that each radial line starting from C ends at the skeleton of C and its length therefore depends on s (and cannot exceed the radius of curvature $1/\kappa$ emanating from convex boundary points).

A.1.4 Final Curve Gradient

Now, if ψ represents the curve C by its zero level set, and ψ is a L^2-signed-distance function, i.e., if $\psi = \psi_C$, we may simplify the gradient from Eq. (219) by noting that

$\|\nabla\psi_C\| = 1$ and $\nabla_\parallel f^* \cdot \mathcal{N} = 0$. Further substituting Eq. (234) into Eq. (219), we obtain

$$\frac{dE^{in}_{\text{aMS}}(C, f^*, \psi_C)}{dt} = \int_C \left(\alpha(I - f^*)^2 + \beta_\perp \|\nabla_\perp f^*\|^2 + \beta_\parallel \|\nabla_\parallel f^*\|^2\right) \frac{\partial C}{\partial t} \cdot \mathcal{N}\, ds + \ldots$$
$$- 2(\beta_\perp - \beta_\parallel) \int_R \left(\frac{\partial C^*}{\partial t} \cdot \mathcal{N}^*\right) \nabla \cdot \left(\frac{\partial f^*}{\partial \eta} \nabla_\parallel f^*\right) dx. \tag{241}$$

Changing coordinates to s_τ and τ, we use the following substitution

$$\nabla \cdot \left(\frac{\partial f^*}{\partial \tau} \nabla_\parallel f^*\right) = \nabla_\parallel \frac{\partial f^*}{\partial \tau} \cdot \nabla_\parallel f^* + \frac{\partial f^*}{\partial \tau} \nabla \cdot \nabla_\parallel f^*$$
$$= \left(\frac{\partial}{\partial s_\tau} \frac{\partial f^*}{\partial \tau}\right) \frac{\partial f^*}{\partial s_\tau} + \frac{\partial f^*}{\partial \tau} \left(\frac{\partial}{\partial s_\tau}\right)^2 f^* = \frac{\partial}{\partial s_\tau} \left(\frac{\partial f^*}{\partial \tau} \frac{\partial f^*}{\partial s_\tau}\right), \tag{242}$$

where the relationship

$$\nabla \cdot \nabla_\parallel = \nabla_\parallel \cdot \nabla_\parallel + \nabla_\perp \cdot \nabla_\parallel = \nabla_\parallel \cdot \nabla_\parallel = \left(\frac{\partial C^*}{\partial s} \frac{\partial}{\partial s_\tau}\right) \cdot \left(\frac{\partial C^*}{\partial s} \frac{\partial}{\partial s_\tau}\right)$$
$$= \underbrace{\frac{\partial C^*}{\partial s} \cdot \left(\frac{\partial}{\partial s_\tau} \frac{\partial C^*}{\partial s}\right)}_{0} \frac{\partial}{\partial s_\tau} + \underbrace{\left(\frac{\partial C^*}{\partial s} \cdot \frac{\partial C^*}{\partial s}\right)}_{1} \left(\frac{\partial}{\partial s_\tau} \frac{\partial}{\partial s_\tau}\right) = \left(\frac{\partial}{\partial s_\tau}\right)^2 \tag{243}$$

was exploited. Finally, with Eq. (235) and Eq. (242), the final gradient with respect to t becomes

$$\frac{dE^{in}_{\text{aMS}}(C, f^*)}{dt} = \int_C \left(\alpha(I - f^*)^2 + \beta_\perp \|\nabla_\perp f^*\|^2 + \beta_\parallel \|\nabla_\parallel f^*\|^2\right) \frac{\partial C}{\partial t} \cdot \mathcal{N}\, ds + \ldots$$
$$-2(\beta_\perp - \beta_\parallel) \int_C \left(\frac{\partial C}{\partial t} \cdot \mathcal{N}\right) \int_{\tau_{\min}(s)}^0 \nabla \cdot \left(\frac{\partial f^*}{\partial \tau} \nabla_\parallel f^*\right) (1 + \kappa^* \tau)\, d\tau\, ds$$
$$= \int_C \left(\alpha(I - f^*)^2 + \beta_\perp \|\nabla_\perp f^*\|^2 + \beta_\parallel \|\nabla_\parallel f^*\|^2\right) \frac{\partial C}{\partial t} \cdot \mathcal{N}\, ds + \ldots$$
$$-2(\beta_\perp - \beta_\parallel) \int_C \left(\frac{\partial C}{\partial t} \cdot \mathcal{N}\right) \int_{\tau_{\min}(s)}^0 \frac{\partial}{\partial s_\tau} \left(\frac{\partial f^*}{\partial \tau} \frac{\partial f^*}{\partial s_\tau}\right) (1 + \kappa^* \tau)\, d\tau\, ds, \tag{244}$$

where $\tau_{\min}(s)$ is the level set value at the skeleton point corresponding to $C(s)$. Therefore, the curve flow (considering only integral terms over R) is given by

$$C^{in}_t = [-\alpha(I - f^*)^2 - \beta_\perp \|\nabla_\perp f^*\|^2 - \beta_\parallel \|\nabla_\parallel f^*\|^2 + \ldots$$
$$+ 2(\beta_\perp - \beta_\parallel) \int_{\tau_{\min}(s)}^0 \frac{\partial}{\partial s_\tau} \left(\frac{\partial f^*}{\partial \tau} \frac{\partial f^*}{\partial s_\tau}\right) (1 + \kappa^* \tau)\, d\tau] \mathcal{N}. \tag{245}$$

A.2 Iso-Contour Limiting Case

Recall that, in the iso-contour limiting case, we have $\nabla_{\|} f_{\text{in}} = 0$ and impose f to be constant along level set curves in the general anisotropic Mumford-Shah functional, i.e.,

$$f(x) = \mathrm{f}(\psi(x)), \tag{246}$$

where $\mathrm{f} : R \to \mathbb{R}$. With this constraint, the interior part of the energy becomes

$$\begin{aligned} E^{in}_{\text{aMS}}(C, \mathrm{f}) &= \int_R \alpha(I - \mathrm{f} \circ \psi)^2 + \beta \|\nabla_{\perp}(\mathrm{f} \circ \psi)\|^2 dx \\ &= \int_R \alpha(I - \mathrm{f} \circ \psi)^2 + \beta \left(\dot{\mathrm{f}} \circ \psi\right)^2 \|\nabla \psi\|^2 dx \\ &= \int_R \alpha I^2 dx + \int_R \alpha \left((\mathrm{f} \circ \psi)^2 - 2I\,(\mathrm{f} \circ \psi)\right) + \beta \left(\dot{\mathrm{f}} \circ \psi\right)^2 \|\nabla \psi\|^2 dx. \end{aligned} \tag{247}$$

after renaming $\beta := \beta_{\perp}$. Here, the zero level set of ψ is the curve C, thus E^{in}_{aMS} does not depend on ψ explicitly. For the derivation of the minimizing equations, however, we will first assume that ψ is an arbitrary level set function, and will later make use of the fact that C is the zero level set of ψ.

A.2.1 Euler-Lagrange ODE for Radiance Function

For an arbitrary fixed ψ and, for a fixed curve C, minimizing $E^{in}_{\text{aMS}}(C, \mathrm{f})$ with respect to f is equivalent to minimizing

$$\hat{E}^{in}_{\text{aMS}}(C, \mathrm{f}, \psi) = \int_R \alpha \left((\mathrm{f} \circ \psi)^2 - 2I\,(\mathrm{f} \circ \psi)\right) + \beta \left(\dot{\mathrm{f}} \circ \psi\right)^2 \|\nabla \psi\|^2 dx \tag{248}$$

since the first term of $E^{in}_{\text{aMS}}(C, \mathrm{f})$ does not depend on f. Now, recall the coarea formula, which expresses the volumetric integral of one function $h : R \mapsto \mathbb{R}$ in terms of the integral along the level sets of another function $\psi : R \mapsto \mathbb{R}$ whose gradient is non-zero almost everywhere

$$\int_R h \|\nabla \psi\| dx = \int_{-\infty}^{+\infty} \left(\int_{C_\tau} h \, ds_\tau \right) d\tau, \tag{249}$$

where $C_\tau = \psi^{-1}(\tau) = \{x \in R : \psi(x) = \tau\}$ denotes the entire τ-level set of ψ and where ds_τ denotes the geometric measure along the τ-level surface (n-1 dimensional

Hausdorff measure). Using the co-area formula we may rewrite $\hat{E}_{in}$ as follows.

$$\begin{aligned}
\hat{E}^{in}_{\text{aMS}}(C,\mathrm{f},\psi) &= \int_R \left(\alpha\frac{(\mathrm{f}\circ\psi)^2 - 2I\,(\mathrm{f}\circ\psi)}{\|\nabla\psi\|} + \beta\left(\dot{\mathrm{f}}\circ\psi\right)^2\|\nabla\psi\|\right)\|\nabla\psi\|dx \\
&= \int_{\tau_{\min}}^{\tau_{\max}}\int_{C_\tau}\left(\alpha\frac{\mathrm{f}^2(\tau) - 2I\,\mathrm{f}(\tau)}{\|\nabla\psi\|} + \beta\dot{\mathrm{f}}^2(\tau)\|\nabla\psi\|\right)ds_\tau\,d\tau \\
&= \int_{\tau_{\min}}^{\tau_{\max}}\left(\alpha\mathrm{f}^2\int_{C_\tau}\frac{1}{\|\nabla\psi\|}ds_\tau - 2\alpha\mathrm{f}\int_{C_\tau}\frac{I}{\|\nabla\psi\|}ds_\tau + \beta\dot{\mathrm{f}}^2\int_{C_\tau}\|\nabla\psi\|ds_\tau\right)d\tau \\
&= \int_{\tau_{\min}}^{\tau_{\max}}\left(\alpha\mathrm{f}^2 m_1(\tau) - 2\alpha\mathrm{f}m_2(\tau) + \beta\dot{\mathrm{f}}^2 m_3(\tau)\right)d\tau, \qquad (250)
\end{aligned}$$

where

$$m_1(\tau) = \int_{C_\tau}\frac{1}{\|\nabla\psi\|}\,ds_\tau, \quad m_2(\tau) = \int_{C_\tau}\frac{I(x)}{\|\nabla\psi\|}\,ds_\tau, \quad m_3(\tau) = \int_{C_\tau}\|\nabla\psi\|\,ds_\tau, \quad (251)$$

and $\tau_{\min} = \min_{x\in R}\psi(x)$ and $\tau_{\max} = \max_{x\in R}\psi(x)$. The Euler-Langrange equations—now in the form of an ODE rather than a PDE compared to the general case—for the optimal one-dimensional function f^* become

$$\alpha\,(m_1\mathrm{f}^* - m_2) - \beta\frac{d(\dot{\mathrm{f}}^* m_3)}{d\tau} = 0, \qquad (252)$$

or equivalently,

$$-\beta m_3\ddot{\mathrm{f}}^* - \beta\dot{m}_3\dot{\mathrm{f}}^* + \alpha m_1\mathrm{f}^* = \alpha m_2 \qquad (253)$$

with vanishing initial and final derivatives $\dot{\mathrm{f}}^*(\tau_{\min})$=$\dot{\mathrm{f}}^*(\tau_{\max})$=0 when $\beta > 0$. If we choose ψ to be the signed-distance function ψ_C to the boundary C, then from the property $\|\nabla\psi_C\| = 1$, the coefficients m_1 and m_3 both match the total Hausdorff measure (arclength, surface area, etc.) of the corresponding τ-level set, while the coefficient m_2 simply becomes the integral of the image I along the corresponding τ-level set.

A.2.1.1 Curve Gradient for Arbitrary Level Set Function

Taking C, f^*, and ψ as functions of artificial time, the total derivative of $E^{in}_{\text{aMS}}(C,\mathrm{f}^*,\psi)$ with respect to t is derived next. We carry out the derivative

$$\frac{E^{in}_{\text{aMS}}(C,\mathrm{f}^*,\psi)}{dt} = \int_C\left(\alpha(I-\mathrm{f}^*\circ\psi)^2 + \beta\|\nabla\psi\|^2\left(\frac{\partial\mathrm{f}^*}{\partial\tau}\circ\psi\right)^2\right)\frac{\partial C}{\partial t}\cdot\mathcal{N}\,ds + \ldots$$

$$+\int_R \frac{\partial}{\partial t}\left(\alpha(I-\mathrm{f}^*\circ\psi)^2+\beta\|\nabla\psi\|^2\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2\right)dx \tag{254}$$

$$\begin{aligned}
&=\int_C\left(\alpha(I-\mathrm{f}^*\circ\psi)^2+\beta\|\nabla\psi\|^2\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2\right)\frac{\partial C}{\partial t}\cdot\mathcal{N}\,ds+\dots\\
&\quad+2\int_R -\alpha(I-\mathrm{f}^*\circ\psi)\frac{\partial \mathrm{f}^*}{\partial t}\circ\psi+\beta\|\nabla\psi\|^2\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\left(\frac{\partial}{\partial t}\frac{\partial \mathrm{f}^*}{\partial\tau}\right)\circ\psi\,dx+\dots\\
&\quad+2\int_R -\alpha(I-\mathrm{f}^*\circ\psi)\frac{\partial\psi}{\partial t}\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi+\beta\|\nabla\psi\|^2\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\frac{\partial\psi}{\partial t}\left(\frac{\partial}{\partial\tau}\frac{\partial \mathrm{f}^*}{\partial\tau}\right)\circ\psi\,dx+\dots\\
&\quad+\int_R \beta\frac{\partial\|\nabla\psi\|^2}{\partial t}\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2dx.
\end{aligned} \tag{255}$$

Application of the coarea formula yields

$$\begin{aligned}
\frac{E^{in}_{\mathrm{aMS}}(C,\mathrm{f}^*,\psi)}{dt}&=\int_C\left(\alpha(I-\mathrm{f}^*\circ\psi)^2+\beta\|\nabla\psi\|^2\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2\right)\frac{\partial C}{\partial t}\cdot\mathcal{N}\,ds+\dots\\
&\quad+2\int_{\tau_{\min}}^{\tau_{\max}}\alpha m_1\mathrm{f}^*\frac{\partial \mathrm{f}^*}{\partial t}-\alpha m_2\frac{\partial \mathrm{f}^*}{\partial t}+\beta m_3\frac{\partial \mathrm{f}^*}{\partial\tau}\left(\frac{\partial}{\partial t}\frac{\partial \mathrm{f}^*}{\partial\tau}\right)d\tau+\dots\\
&\quad+2\int_R\frac{\partial\psi}{\partial t}\left(-\alpha(I-\mathrm{f}^*\circ\psi)\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi+\beta\|\nabla\psi\|^2\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\left(\frac{\partial}{\partial\tau}\frac{\partial \mathrm{f}^*}{\partial\tau}\right)\circ\psi\right)dx+\dots\\
&\quad+\int_R\beta\frac{\partial\|\nabla\psi\|^2}{\partial t}\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2dx.
\end{aligned} \tag{256}$$

Integration by parts of the second row results in

$$\begin{aligned}
\frac{E^{in}_{\mathrm{aMS}}(C,\mathrm{f}^*,\psi)}{dt}&=\int_C\left(\alpha(I-\mathrm{f}^*\circ\psi)^2+\beta\|\nabla\psi\|^2\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2\right)\frac{\partial C}{\partial t}\cdot\mathcal{N}\,ds+\dots\\
&\quad+2\int_{\tau_{\min}}^{\tau_{\max}}\frac{\partial \mathrm{f}^*}{\partial t}\left(\alpha\left(m_1\mathrm{f}^*-m_2\right)-\beta\frac{\partial}{\partial\tau}\left(m_3\frac{\partial \mathrm{f}^*}{\partial\tau}\right)\right)d\tau+2\beta\left[m_3\frac{\partial \mathrm{f}^*}{\partial\tau}\frac{\partial \mathrm{f}^*}{\partial t}\right]_{\tau_{\min}}^{\tau_{\max}}+\dots\\
&\quad+2\int_R\frac{\partial\psi}{\partial t}\left(-\alpha(I-\mathrm{f}^*\circ\psi)\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi+\beta\|\nabla\psi\|^2\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\left(\frac{\partial}{\partial\tau}\frac{\partial \mathrm{f}^*}{\partial\tau}\right)\circ\psi\right)dx+\dots\\
&\quad+\int_R\beta\frac{\partial\|\nabla\psi\|^2}{\partial t}\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2dx,
\end{aligned} \tag{257}$$

and it is observed that the second row vanishes due to the first-order necessary condition on f^* Eq. (253) plus boundary conditions. Further noting that $\frac{\partial\|\nabla\psi\|^2}{\partial t}=0$ if ψ is enforced to maintain a signed-distance function, we arrive at the total derivative

$$\begin{aligned}
\frac{E^{in}_{\mathrm{aMS}}(C,\mathrm{f}^*,\psi)}{dt}&=\int_C\left(\alpha(I-\mathrm{f}^*\circ\psi)^2+\beta\|\nabla\psi\|^2\left(\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\right)^2\right)\frac{\partial C}{\partial t}\cdot\mathcal{N}\,ds+\dots\\
&\quad+2\int_R\frac{\partial\psi}{\partial t}\left(-\alpha(I-\mathrm{f}^*\circ\psi)\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi+\beta\|\nabla\psi\|^2\frac{\partial \mathrm{f}^*}{\partial\tau}\circ\psi\left(\frac{\partial}{\partial\tau}\frac{\partial \mathrm{f}^*}{\partial\tau}\right)\circ\psi\right)dx.
\end{aligned} \tag{258}$$

A.2.1.2 Final Curve Gradient

Finally, by connecting the level set function with the boundary curve, i.e., by substituting $\psi = \psi_C$, may use results from Appendix A.1 that $\frac{\partial \psi_C}{\partial t} = -\frac{\partial C^*}{\partial t} \cdot \mathcal{N}^*$ to write Eq. (258) as

$$\begin{aligned}
\frac{E^{in}_{\text{aMS}}(C, \mathrm{f}^*)}{dt} &= \int_C \left(\alpha \left(I - \mathrm{f}^*(0) \right)^2 + \beta \left(\frac{\partial \mathrm{f}^*}{\partial \tau}(0) \right)^2 \right) \frac{\partial C}{\partial t} \cdot \mathcal{N} \, ds + \dots \\
&\quad - 2 \int_R \left(\frac{\partial C^*}{\partial t} \cdot \mathcal{N}^* \right) \left(-\alpha (I - \mathrm{f}^* \circ \psi) \frac{\partial \mathrm{f}^*}{\partial \tau} \circ \psi + \beta \frac{\partial \mathrm{f}^*}{\partial \tau} \circ \psi \left(\frac{\partial}{\partial \tau} \frac{\partial \mathrm{f}^*}{\partial \tau} \right) \circ \psi \right) dx \\
&= \int_C \left(\alpha \left(I - \mathrm{f}^*(0) \right)^2 + \beta \left(\frac{\partial \mathrm{f}^*}{\partial \tau}(0) \right)^2 \right) \frac{\partial C}{\partial t} \cdot \mathcal{N} \, ds + \dots \\
&\quad - 2 \int_C \int_{\tau_{\min}(s)}^0 \left(\frac{\partial C^*}{\partial t} \cdot \mathcal{N}^* \right) \left(-\alpha (I - \mathrm{f}^*) \frac{\partial \mathrm{f}^*}{\partial \tau} + \beta \frac{\partial \mathrm{f}^*}{\partial \tau} \left(\frac{\partial}{\partial \tau} \frac{\partial \mathrm{f}^*}{\partial \tau} \right) \right) (1 + \tau \kappa^*) \, d\tau \, ds \\
&= \int_C \left(\alpha \left(I - \mathrm{f}^*(0) \right)^2 \right) \frac{\partial C}{\partial t} \cdot \mathcal{N} \, ds + \dots \\
&\quad - 2 \int_C \left(\frac{\partial C}{\partial t} \cdot \mathcal{N} \right) \left(\int_{\tau_{\min}(s)}^0 \frac{\partial \mathrm{f}^*}{\partial \tau} \left(-\alpha (I - \mathrm{f}^*) + \beta \frac{\partial}{\partial \tau} \frac{\partial \mathrm{f}^*}{\partial \tau} \right) (1 + \tau \kappa^*) \, d\tau \right) ds \quad (259)
\end{aligned}$$

since $\frac{\partial \mathrm{f}^*}{\partial \tau}(0) = 0$ by the boundary conditions on f^*. Hence, the gradient descent curve evolution (arising from interior integrals in the aMS iso-contour energy) is given by

$$C^{in}_t = \left[-\alpha \left(I - \mathrm{f}^*(0) \right)^2 + 2 \int_{\tau_{\min}(s)}^0 \frac{\partial \mathrm{f}^*}{\partial \tau} \left(-\alpha (I - \mathrm{f}^*) + \beta \frac{\partial}{\partial \tau} \frac{\partial \mathrm{f}^*}{\partial \tau} \right) (1 + \tau \kappa^*) \, d\tau \right] \mathcal{N}. \quad (260)$$

A.3 Shape-Plus-Radiance Priors Case

A.3.1 Assumptions and Preliminary Calculations

Shape model. In this derivation we assume that the level set function Φ is represented by a linear combination of given shape components $\Phi_1, \dots, \Phi_K$, and thus depends on a finite-dimensional vector of shape parameter weights $\boldsymbol{w} = (w_1, \dots, w_K)$ as follows

$$\Phi(x; \boldsymbol{w}) = \bar{\Phi}(x) + \sum_{i=1}^{K} w_i \Phi_i(x), \quad (261)$$

where $\bar{\Phi}$ represents the pointwise mean of the aligned training set of signed distance functions. Note that $\bar{\Phi}$ will generally not be a signed distance function itself. In our

specific application, we take $\Phi_1, \ldots, \Phi_K$ to be the first K principal components from the PCA analysis carried out on a set of aligned training signed-distance functions as explained in Section 3.3.2.

Pose parameters. Further, a set of L pose parameters $\boldsymbol{p} = (p_1, \ldots, p_L)$ is added to define an invertible shape-independent domain transformation $g(x; \boldsymbol{p})$ (such as translation, rotation, and scaling) which maps any given spatial location x to $\hat{x} = g(x; \boldsymbol{p})$ and whose inverse mapping in turn maps each such location $\hat{x}$ back to its original location $x = g^{-1}(\hat{x}; \boldsymbol{p})$. The domain-transformed level set function $\hat{\Phi}$ is then defined as follows

$$\hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) = \Phi\left(g^{-1}(\hat{x}; \boldsymbol{p}); \boldsymbol{w}\right). \tag{262}$$

By denoting the Jacobian of the forward mapping as $\frac{\partial g}{\partial x}$ and differentiating $\Phi(x; \boldsymbol{w}) = \hat{\Phi}\left(g(x; \boldsymbol{p}); \boldsymbol{p}, \boldsymbol{w}\right)$ in x, we may relate the spatial gradient ∇_x of Φ within the original (training) domain to the spatial gradient $\nabla_{\hat{x}}$ of $\hat{\Phi}$ within the transformed domain as follows

$$\nabla_x \Phi(x; \boldsymbol{w}) = \left[\frac{\partial g}{\partial x}(x; \boldsymbol{p})\right]^T \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}). \tag{263}$$

Differentiating instead with respect to one of the pose parameters $p_1, \ldots, p_L$ in turn yields the following expression

$$\frac{\partial \hat{\Phi}}{\partial p_l}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) = -\nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) \cdot \frac{\partial g}{\partial p_l}(x; \boldsymbol{p}), \tag{264}$$

whereas differentiating with respect to one of the shape parameters $w_1, \ldots, w_K$ gives the expression

$$\frac{\partial \hat{\Phi}}{\partial w_i}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) = \frac{\partial \Phi}{\partial w_i}(x; \boldsymbol{w}) = \Phi_i(x). \tag{265}$$

Normal speed of zero level set. Differentiating the relationship

$$\hat{\Phi}\left(\hat{C}(\boldsymbol{p}, \boldsymbol{w}), \boldsymbol{p}, \boldsymbol{w}\right) = 0 \tag{266}$$

with respect to any of the shape parameters w_i yields

$$\frac{\partial \hat{\Phi}}{\partial w_i} + \nabla_{\hat{x}} \hat{\Phi} \cdot \frac{\partial \hat{C}}{\partial w_i} = 0 \tag{267}$$

If we now substitute $\nabla_{\hat{x}} \hat{\Phi} = \hat{\mathcal{N}} \|\nabla_{\hat{x}} \hat{\Phi}\|$, we obtain

$$\frac{\partial \hat{C}}{\partial w_i} \cdot \hat{\mathcal{N}} = -\frac{1}{\|\nabla_{\hat{x}} \hat{\Phi}\|} \frac{\partial \hat{\Phi}}{\partial w_i} = -\frac{\Phi_i(x)}{\|\nabla_{\hat{x}} \hat{\Phi}\|}. \tag{268}$$

Gradient norm derivatives. Differentiating the relationship Eq. (263) with respect to one of the pose parameters p_l

$$\frac{d}{dp_l} \left(\nabla_x \Phi(x; \boldsymbol{w}) \right) = \frac{d}{dp_l} \left(\left[\frac{\partial g}{\partial x}(x; \boldsymbol{p}) \right]^T \nabla_{\hat{x}} \hat{\Phi}(g(x; \boldsymbol{p}); \boldsymbol{p}, \boldsymbol{w}) \right) \tag{269}$$

yields

$$0 = \frac{d}{dp_l} \left(\left[\frac{\partial g}{\partial x}(x; \boldsymbol{p}) \right]^T \right) \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) + \left[\frac{\partial g}{\partial x}(x; \boldsymbol{p}) \right]^T \frac{d}{dp_l} \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}), \tag{270}$$

which after carrying out the derivatives results in

$$\begin{aligned} 0 = & \left[\frac{\partial^2 g}{\partial x \partial p_l}(x; \boldsymbol{p}) \right]^T \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) + \dots \\ & + \left[\frac{\partial g}{\partial x}(x; \boldsymbol{p}) \right]^T \left(\frac{\partial}{\partial p_l} \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) + H_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) \frac{\partial g}{\partial p_l}(x; \boldsymbol{p}) \right), \end{aligned} \tag{271}$$

where $H_{\hat{x}} \hat{\Phi}$ is the Hessian of $\hat{\Phi}$. Therefore,

$$\begin{aligned} \frac{\partial}{\partial p_l} \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) = & -H_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}) \frac{\partial g}{\partial p_l}(x; \boldsymbol{p}) + \\ & - \left[\frac{\partial g}{\partial x}(x; \boldsymbol{p}) \right]^{-T} \left[\frac{\partial^2 g}{\partial x \partial p_l}(x; \boldsymbol{p}) \right]^T \nabla_{\hat{x}} \hat{\Phi}(\hat{x}; \boldsymbol{p}, \boldsymbol{w}), \end{aligned} \tag{272}$$

which in turn allows us to write

$$\frac{\partial}{\partial p_l} \left\| \nabla_{\hat{x}} \hat{\Phi} \right\| = \frac{\nabla_{\hat{x}} \hat{\Phi} \cdot \frac{\partial}{\partial p_l} \nabla_{\hat{x}} \hat{\Phi}}{\left\| \nabla_{\hat{x}} \hat{\Phi} \right\|} = -\frac{\nabla_{\hat{x}}^T \hat{\Phi} \left(H_{\hat{x}} \hat{\Phi} \frac{\partial g}{\partial p_l} + \left[\frac{\partial g}{\partial x} \right]^{-T} \left[\frac{\partial^2 g}{\partial x \partial p_l} \right]^T \nabla_{\hat{x}} \hat{\Phi} \right)}{\left\| \nabla_{\hat{x}} \hat{\Phi} \right\|} \tag{273}$$

or equivalently,

$$\left\| \nabla_{\hat{x}} \hat{\Phi} \right\| \frac{\partial}{\partial p_l} \left\| \nabla_{\hat{x}} \hat{\Phi} \right\| = -\nabla_{\hat{x}}^T \hat{\Phi} \left(H_{\hat{x}} \hat{\Phi} \frac{\partial g}{\partial p_l} + \left[\frac{\partial^2 g}{\partial x \partial p_l} \right] \left[\frac{\partial g}{\partial x} \right]^{-1} \nabla_{\hat{x}} \hat{\Phi} \right). \tag{274}$$

By differentiating Eq. (263) with respect to one of the shape parameters w_i, we obtain

$$\begin{aligned}\nabla_x \Phi_i(x) = \frac{d}{dw_i}\nabla_x \Phi(x;\boldsymbol{w}) &= \frac{d}{dw_i}\left(\left[\frac{\partial g}{\partial x}(x;\boldsymbol{p})\right]^T \nabla_{\hat{x}}\hat{\Phi}(g(x;\boldsymbol{p});\boldsymbol{p},\boldsymbol{w})\right) \\ &= \left[\frac{\partial g}{\partial x}(x;\boldsymbol{p})\right]^T \frac{\partial}{\partial w_i}\nabla_{\hat{x}}\hat{\Phi}(\hat{x};\boldsymbol{p},\boldsymbol{w}) \end{aligned} \tag{275}$$

and so,

$$\frac{\partial}{\partial w_i}\nabla_{\hat{x}}\hat{\Phi}(\hat{x};\boldsymbol{p},\boldsymbol{w}) = \left[\frac{\partial g}{\partial x}(x;\boldsymbol{p})\right]^{-T} \nabla_x \Phi_i(x). \tag{276}$$

Hence,

$$\frac{\partial}{\partial w_i}\left\|\nabla_{\hat{x}}\hat{\Phi}\right\| = \frac{\nabla_{\hat{x}}\hat{\Phi}\cdot\frac{\partial}{\partial w_i}\nabla_{\hat{x}}\hat{\Phi}}{\left\|\nabla_{\hat{x}}\hat{\Phi}\right\|} = \frac{\nabla_{\hat{x}}^T\hat{\Phi}\left[\frac{\partial g}{\partial x}\right]^{-T}\nabla_x\Phi_i}{\left\|\nabla_{\hat{x}}\hat{\Phi}\right\|}, \tag{277}$$

and finally,

$$\left\|\nabla_{\hat{x}}\hat{\Phi}\right\|\frac{\partial}{\partial w_i}\left\|\nabla_{\hat{x}}\hat{\Phi}\right\| = \nabla_x^T\Phi_i\left[\frac{\partial g}{\partial x}\right]^{-1}\nabla_{\hat{x}}\hat{\Phi}. \tag{278}$$

Radiance model. We further assume that the radiance function F is also dependent on a finite-dimensional vector of radiance parameters $\boldsymbol{v} = (v_1, \ldots, v_M)$ as follows

$$\mathrm{F}(\tau;\boldsymbol{v}) = \bar{\mathrm{F}}(\tau) + \sum_{i=1}^{M} v_i \mathrm{F}_i(\tau), \tag{279}$$

where $\bar{\mathrm{F}}$ represents the point-wise mean of the training set's one-dimensional radiance functions and where $\mathrm{F}_1, \ldots, \mathrm{F}_M$ denote the first M principal components from the PCA analysis carried out on this set of radiance functions. In this specific application, these one-dimensional functions are the photo-geometric representations of objects as defined in Section 3.3.1, and τ will, therefore, be replaced by the object's level set function.

A.3.2 Energy Formulation

We may now write the interior part of the energy functional in terms of the pose, shape, and radiance parameters as follows

$$E^{in}_{\text{aMS-SR}}(\boldsymbol{w},\boldsymbol{v},\boldsymbol{p}) = \alpha\int_{\hat{R}(\boldsymbol{p},\boldsymbol{w})}\left(I(\hat{x}) - \mathrm{F}\left(\hat{\Phi}(\hat{x};\boldsymbol{p},\boldsymbol{w});\boldsymbol{v}\right)\right)^2 d\hat{x} + \ldots$$

$$+ \beta \int_{\hat{R}(\boldsymbol{p},\boldsymbol{w})} \left(\frac{\partial \mathrm{F}}{\partial \tau} \left(\hat{\Phi}(\hat{x};\boldsymbol{p},\boldsymbol{w}); \boldsymbol{v} \right) \left\| \nabla_{\hat{x}} \hat{\Phi}(\hat{x};\boldsymbol{p},\boldsymbol{w}) \right\| \right)^2 d\hat{x}, \qquad (280)$$

where $\hat{R}(\boldsymbol{p},\boldsymbol{w})$ denotes the spatially transformed set $R(\boldsymbol{w}) = \{x : \Phi(x;\boldsymbol{w}) < 0\}$ associated with the pose parameters $\boldsymbol{p}$, or more formally, $\hat{R}(\boldsymbol{p},\boldsymbol{w}) = \{\hat{x} : g^{-1}(\hat{x};\boldsymbol{p}) \in R(\boldsymbol{w})\}$. In a similar way, if we let $C(\boldsymbol{w})$ denote the zero level set of $\Phi(x;\boldsymbol{w})$ with unit outward normal and arclength parameter denoted by $\mathcal{N}(\boldsymbol{w})$ and $s(\boldsymbol{w})$ respectively, then $\hat{C}(\boldsymbol{p},\boldsymbol{w})$ denotes the zero level set of $\hat{\Phi}(\hat{x};\boldsymbol{p},\boldsymbol{w})$ with unit outward normal and arclength parameter denoted by $\hat{\mathcal{N}}(\boldsymbol{p},\boldsymbol{w})$ and $\hat{s}(\boldsymbol{p},\boldsymbol{w})$.

A.3.3 Energy Derivatives

For the sake of compactness we will stop explicitly showing parameter dependencies for the level set and radiance functions. In addition, the spatial arguments of I and $\hat{\Phi}$ will match the spatial integration variable $\hat{x}$ whereas the spatial variable of the domain transformation mapping g as well as the pre-transformed level set function Φ and its additive principal components Φ_i will match the inverse-mapped integration variable $x = g^{-1}(\hat{x};\boldsymbol{p})$. The corresponding gradient operators ∇_x and $\nabla_{\hat{x}}$ will also be written without the subscripts x and $\hat{x}$ but will be understood in the same context just outlined. In other words, quantities in the image domain, where segmentation is performed, depend on $\hat{x}$, whereas quantities in the training domain, where shape and radiance PCA was performed, depend on x. We will continue, however, to explicitly show the one-dimensional argument τ of the radiance function F to emphasize its dependence on the level set value $\hat{\Phi}$ at $\hat{x}$ rather than $\hat{x}$ itself.

Differentiating with respect to one of a pose parameter p_l yields

$$\begin{aligned}
\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial p_l} = & \int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0) \right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \left\| \nabla \hat{\Phi} \right\| \right)^2 \right) \left(\frac{\partial \hat{C}}{\partial p_l} \cdot \hat{\mathcal{N}} \right) d\hat{s} + \ldots \\
& + 2\alpha \int_{\hat{R}} \left(I - \mathrm{F}(\hat{\Phi}) \right) \left(-\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \frac{\partial \hat{\Phi}}{\partial p_l} \right) d\hat{x} + \ldots \\
& + 2\beta \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\| \left(\frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\| \frac{\partial \hat{\Phi}}{\partial p_l} + \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \frac{\partial}{\partial p_l} \left\| \nabla \hat{\Phi} \right\| \right) d\hat{x}, \qquad (281)
\end{aligned}$$

which can be simplified by using Eq. (264) and Eq. (274) in the second and third line to obtain

$$\begin{aligned}\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial p_l} &= \int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0) \right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \left\| \nabla \hat{\Phi} \right\| \right)^2 \right) \left(\frac{\partial \hat{C}}{\partial p_l} \cdot \hat{\mathcal{N}} \right) d\hat{s} + \dots \\ &+ 2\alpha \int_{\hat{R}} \left(I - \mathrm{F}(\hat{\Phi}) \right) \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \nabla \hat{\Phi} \cdot \frac{\partial g}{\partial p_l} d\hat{x} + \dots \\ &- 2\beta \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\| \left(\frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\| \nabla \hat{\Phi} \cdot \frac{\partial g}{\partial p_l} \right) d\hat{x} + \dots \\ &- 2\beta \int_{\hat{R}} \left(\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \right)^2 \nabla \hat{\Phi} \cdot \left(H\hat{\Phi} \frac{\partial g}{\partial p_l} + \left[\frac{\partial^2 g}{\partial x \partial p_l} \right] \left[\frac{\partial g}{\partial x} \right]^{-1} \nabla \hat{\Phi} \right) d\hat{x}, \end{aligned} \tag{282}$$

which after rearranging terms finally results in

$$\begin{aligned}\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial p_l} &= \int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0) \right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \left\| \nabla \hat{\Phi} \right\| \right)^2 \right) \left(\frac{\partial \hat{C}}{\partial p_l} \cdot \hat{\mathcal{N}} \right) d\hat{s} + \dots \\ &+ 2 \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \left(\nabla \hat{\Phi} \cdot \frac{\partial g}{\partial p_l} \right) \left(\alpha \left(I - \mathrm{F}(\hat{\Phi}) \right) - \beta \frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\|^2 \right) d\hat{x} + \dots \\ &- 2\beta \int_{\hat{R}} \left(\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \right)^2 \nabla \hat{\Phi} \cdot \left(H\hat{\Phi} \frac{\partial g}{\partial p_l} + \left[\frac{\partial^2 g}{\partial x \partial p_l} \right] \left[\frac{\partial g}{\partial x} \right]^{-1} \nabla \hat{\Phi} \right) d\hat{x}. \end{aligned} \tag{283}$$

Differentiating with respect to one of the shape parameters w_i yields

$$\begin{aligned}\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial w_i} &= \int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0) \right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \left\| \nabla \hat{\Phi} \right\| \right)^2 \right) \left(\frac{\partial \hat{C}}{\partial w_i} \cdot \hat{\mathcal{N}} \right) d\hat{s} + \dots \\ &+ 2\alpha \int_{\hat{R}} \left(I - \mathrm{F}(\hat{\Phi}) \right) \left(-\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \frac{\partial \hat{\Phi}}{\partial w_i} \right) d\hat{x} + \dots \\ &+ 2\beta \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\| \left(\frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\| \frac{\partial \hat{\Phi}}{\partial w_i} + \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \frac{\partial}{\partial w_i} \left\| \nabla \hat{\Phi} \right\| \right) d\hat{x}, \end{aligned} \tag{284}$$

which is simplified by using Eq. (268) in line one, Eq. (265) in lines two and three, and Eq. (278) in line three to obtain

$$\begin{aligned}\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial w_i} &= -\int_{\hat{C}} \left(\alpha \left(I - \mathrm{F}(0) \right)^2 + \beta \left(\frac{\partial \mathrm{F}}{\partial \tau}(0) \left\| \nabla \hat{\Phi} \right\| \right)^2 \right) \frac{\Phi_i}{\| \nabla \hat{\Phi} \|} d\hat{s} + \dots \\ &- 2 \int_{\hat{R}} \frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \Phi_i \left(\alpha \left(I - \mathrm{F}(\hat{\Phi}) \right) - \beta \frac{\partial^2 \mathrm{F}}{\partial \tau^2}(\hat{\Phi}) \left\| \nabla \hat{\Phi} \right\|^2 \right) d\hat{x} + \dots \\ &+ 2\beta \int_{\hat{R}} \left(\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi}) \right)^2 \nabla \Phi_i \cdot \left(\left[\frac{\partial g}{\partial x} \right]^{-1} \nabla \hat{\Phi} \right) d\hat{x}. \end{aligned} \tag{285}$$

Differentiating with respect to one of the radiance parameters v_i yields

$$\begin{aligned}\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial v_i} &= 2\int_{\hat{R}} -\alpha\left(I(\hat{x}) - \mathrm{F}(\hat{\Phi})\right)\frac{\partial \mathrm{F}}{\partial v_i}(\hat{\Phi}) + \beta\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi})\left\|\nabla\hat{\Phi}\right\|^2\frac{\partial^2 \mathrm{F}}{\partial\tau\partial v_i}(\hat{\Phi})\,d\hat{x} \\ &= 2\int_{\hat{R}} -\alpha\left(I(\hat{x}) - \mathrm{F}(\hat{\Phi})\right)F_i(\hat{\Phi}) + \beta\frac{\partial \mathrm{F}}{\partial \tau}(\hat{\Phi})\left\|\nabla\hat{\Phi}\right\|^2\frac{d\mathrm{F}_i}{d\tau}(\hat{\Phi})\,d\hat{x}. \qquad (286)\end{aligned}$$

These derivatives are used for gradient descent to iteratively minimize the anisotropic Mumford-Shah with shape and radiance priors Eq. (280). In case coupled shape-plus-radiance PCA components are used to model shape and radiance, we have that

$$\frac{\partial E^{in}_{\text{aMS-SR}}}{\partial l_i} = \frac{\partial E^{in}_{\text{aMS-SR}}}{\partial w_i} + \frac{\partial E^{in}_{\text{aMS-SR}}}{\partial v_i}, \qquad (287)$$

where $l_1, \ldots, l_K$ are the weight parameters for the coupled shape-plus-radiance PCA components.

APPENDIX B

SHAPE DERIVATIVES FOR ADJOINT ACTIVE CONTOURS

B.1 Acoustics

Referring to the setup from Section 4.2, this appendix makes some simplifications to the force term presented in [29]. We start with the shape gradient for inclusion l given by

$$\dot{\Gamma}^l = F^l \mathcal{N}^l \tag{288}$$

where

$$\begin{aligned} F^l &= d^l \left(\frac{\partial w^l}{\partial \mathcal{N}^l} + \frac{\partial w^0}{\partial \mathcal{N}^0} \right) \frac{\partial u^l}{\partial \mathcal{N}^l} - d^0 \left(\frac{\partial u^l}{\partial \mathcal{N}^l} + \frac{\partial u^0}{\partial \mathcal{N}^0} \right) \frac{\partial w^0}{\partial \mathcal{N}^0} + ... \\ &\quad + d^0 \nabla w^0 \cdot \nabla u^0 - d^l \nabla w^l \cdot \nabla u^l + (k^0 - k^l) w^l u^l \\ &= d^l \left(\frac{\partial w^l}{\partial \mathcal{N}^l} - \frac{\partial w^0}{\partial \mathcal{N}^l} \right) \frac{\partial u^l}{\partial \mathcal{N}^l} + d^0 \left(\frac{\partial u^l}{\partial \mathcal{N}^l} - \frac{\partial u^0}{\partial \mathcal{N}^l} \right) \frac{\partial w^0}{\partial \mathcal{N}^l} + ... \\ &\quad + d^0 \nabla w^0 \cdot \nabla u^0 - d^l \nabla w^l \cdot \nabla u^l + (k^0 - k^l) w^l u^l \end{aligned} \tag{289}$$

as presented in [29]. Observe that

$$\begin{aligned} \left(\frac{\partial w^l}{\partial \mathcal{N}^l} - \frac{\partial w^0}{\partial \mathcal{N}^l} \right) \frac{\partial u^l}{\partial \mathcal{N}^l} &= \left(\frac{\partial w^l}{\partial \mathcal{N}^l} - \frac{\partial w^0}{\partial \mathcal{N}^l} \right) \mathcal{N}^l \cdot \nabla u^l \\ &= \left[\left(\frac{\partial w^l}{\partial \mathcal{N}^l} - \frac{\partial w^0}{\partial \mathcal{N}^l} \right) \mathcal{N}^l + \left(\frac{\partial w^l}{\partial \mathcal{T}^l} - \frac{\partial w^0}{\partial \mathcal{T}^l} \right) \mathcal{T}^l \right] \cdot \nabla u^l \\ &= \left(\nabla w^l - \nabla w^0 \right) \cdot \nabla u^l \end{aligned} \tag{290}$$

since $w^l = w^0$ on Γ^l and so $\partial w^l / \partial \mathcal{T}^l = \partial w^0 / \partial \mathcal{T}^l$, where $\mathcal{T}^l$ is the tangent vector along the interface l. Similarly,

$$\left(\frac{\partial u^l}{\partial \mathcal{N}^l} - \frac{\partial u^0}{\partial \mathcal{N}^l} \right) \frac{\partial w^0}{\partial \mathcal{N}^l} = \left(\nabla u^l - \nabla u^0 \right) \cdot \nabla w^0. \tag{291}$$

Therefore, Eq. (289) reduces to

$$\begin{aligned} F^l &= d^l \left(\nabla w^l - \nabla w^0\right) \cdot \nabla u^l + d^0 \left(\nabla u^l - \nabla u^0\right) \cdot \nabla w^0 + ... \\ &\quad + d^0 \nabla w^0 \cdot \nabla u^0 - d^l \nabla w^l \cdot \nabla u^l + (k^0 - k^l) w^l u^l \\ &= d^0 \nabla u^l \cdot \nabla w^0 - d^l \nabla u^l \cdot \nabla w^0 + (k^0 - k^l) w^l u^l. \\ &= (d^0 - d^l) \nabla u^l \cdot \nabla w^0 + (k^0 - k^l) w^l u^l \end{aligned} \tag{292}$$

Thus,

$$F^l = (d^0 - d^l) \nabla u^l \cdot \nabla w^0 + (k^0 - k^l) w^l u^l. \tag{293}$$

B.2 Elastodynamics

In this appendix, we derive the minimizer for the inclusion setup in Section 4.2 for the misfit energy

$$E(\Gamma^1, ..., \Gamma^L) = \frac{1}{2} \sum_{r=1}^{R} \|u_i(x_r) - \hat{u}_i(x_r)\|^2, \tag{294}$$

where the displacements u_i satisfy the boundary value problem

$$T^{(u^l)}_{ij,j} + \gamma^l u^l_i = S^l_i \quad \text{in } \Omega^l,\ l = 0, ..., L, \tag{295a}$$

$$T^{(u^0)}_{ij} \mathcal{N}^0_j = \bar{t}_i \quad \text{on } \bar{\Gamma}\ , \tag{295b}$$

$$u^l_i - u^0_i = 0 \quad \text{on } \Gamma^l,\ l = 1, ..., L, \tag{295c}$$

$$(T^{(u^l)}_{ij} - T^{(u^0)}_{ij}) \mathcal{N}^l_j = 0 \quad \text{on } \Gamma^l,\ l = 1, ..., L. \tag{295d}$$

Here, u^l_i is u_i restricted to Ω^l, $\gamma^l = \rho^l \omega^2$ where ρ^l is density and ω is angular frequency. S^l_i is a given source term, and $\mathcal{N}^l_i$ is the outward normal vector for Ω^l. For reference, the relation between stress and displacement is stated as

$$T^{(u^l)}_{ij} = \lambda^l u^l_{k,k} \delta_{ij} + \mu^l (u^l_{i,j} + u^l_{j,i}), \tag{296}$$

where λ^l is Lamé's first parameter and μ^l is the Lamé's second parameter, also known as shear modulus. Hence, Eq. (295a) can be written in terms of displacements as

$$(\lambda^l + \mu^l) u^l_{k,ki} + \mu^l u^l_{i,kk} + \gamma^l u^l_i = S^l_i \quad \text{in } \Omega^l,\ l = 0, ..., L. \tag{297}$$

Now, define

$$g(x) = \frac{1}{2}\sum_{r=1}^{R} \|u_i(x_r) - \hat{u}_i(x_r)\|^2 \delta(x - x_r) \tag{298}$$

to write the energy as an integral, and also expand the energy by integrals over the PDEs Eq. (295a) weighted by Lagrange-multiplier functions w_i^l

$$E = \int_{\bar{\Gamma}} g(x)\, d\Gamma + \sum_{l=0}^{L} \int_{\Omega^l} (T_{ij,j}^{(u^l)} + \gamma^l u_i^l - S_i^l) w_i^l \, d\Omega. \tag{299}$$

As a above, w_i^l means w_i restricted to Ω^l, and observe that the second term is zero independent of w_i as the PDEs equal zero. By the adjoint method, we will find defining equations for w_i that simplify the problem, and we will call w_i the *adjoint* displacement field. Integration by parts on the stress terms yields

$$E = \int_{\bar{\Gamma}} g(x)\, d\Gamma + \sum_{l=0}^{L} \int_{\Omega^l} -T_{ij}^{(u^l)} w_{i,j}^l + (\gamma^l u_i^l - S_i^l) w_i^l \, d\Omega + \sum_{l=0}^{L} \int_{\Gamma^l} T_{ij}^{(u^l)} w_i^l \mathcal{N}_j^l \, d\Gamma. \tag{300}$$

Now, writing out $\Gamma^0 = \Gamma \cup \bigcup_{l=1}^{L} \Gamma^l$ in the last integral and using that $\mathcal{N}_i^0 = -\mathcal{N}_i^l$, $l = 1, ..., L$ results in

$$\begin{aligned} E = &\int_{\bar{\Gamma}} g(x)\, d\Gamma + \sum_{l=0}^{L} \int_{\Omega^l} -T_{ij}^{(u^l)} w_{i,j}^l + (\gamma^l u_i^l - S_i^l) w_i^l \, d\Omega + ... \\ &+ \sum_{l=1}^{L} \int_{\Gamma^l} (T_{ij}^{(u^l)} w_i^l - T_{ij}^{(u^0)} w_i^0) \mathcal{N}_j^l \, d\Gamma + \int_{\bar{\Gamma}} T_{ij}^{(u^0)} \mathcal{N}_j^0 w_i^0 \, d\Gamma. \end{aligned} \tag{301}$$

Now, imposing that

$$w_i^0 = w_i^l \quad \text{on } \Gamma^l,\ l = 1, ..., L, \tag{302}$$

we observe that the second to last integral vanishes due to Eq. (295d). Thus, writing out the stress in terms of displacement according to Eq. (296) and using Eq. (295b), we are left with

$$E = \int_{\bar{\Gamma}} g(x) + \bar{t}_i w_i^0 \, d\Gamma + \sum_{l=0}^{L} \int_{\Omega^l} -\lambda^l u_{k,k}^l w_{i,i}^l - \mu^l u_{i,j}^l w_{i,j}^l - \mu^l u_{j,i}^l w_{i,j}^l + (\gamma^l u_i^l - S_i^l) w_i^l \, d\Omega. \tag{303}$$

We now let Γ^l, $l = 1, ..., L$ depend on time t, so u_i^l and w_i^l become functions of t as well, since the displacement fields depend on the interfaces Γ^l. Denoting the partial

time derivative as $(\dot{\bullet})$ and carrying out derivatives with respect to t, we have that

$$\begin{aligned}\frac{dE}{dt} = &\int_{\bar{\Gamma}} \frac{dg(x)}{du_i}\dot{u}_i^0 + \bar{t}_i\dot{w}_i^0\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Gamma^l}(-\lambda^l u^l_{k,k}w^l_{i,i} - \mu^l u^l_{i,j}w^l_{i,j} - \mu^l u^l_{j,i}w^l_{i,j} + (\gamma^l u^l_i - S^l_i)w^l_i)(\dot{\Gamma}^l_m\mathcal{N}^l_m)\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Omega^l} -\lambda^l u^l_{k,k}\dot{w}^l_{i,i} - \mu^l u^l_{i,j}\dot{w}^l_{i,j} - \mu^l u^l_{j,i}\dot{w}^l_{i,j} + (\gamma^l u^l_i - S^l_i)\dot{w}^l_i\, d\Omega + ... \\ &+ \sum_{l=0}^{L}\int_{\Omega^l} -\lambda^l \dot{u}^l_{k,k}w^l_{i,i} - \mu^l \dot{u}^l_{i,j}w^l_{i,j} - \mu^l \dot{u}^l_{j,i}w^l_{i,j} + \gamma^l \dot{u}^l_i w^l_i\, d\Omega. \end{aligned} \tag{304}$$

Performing again integration by parts in the last two integrals on the terms involving both time and spatial derivatives of u_i or w_i, we get

$$\begin{aligned}\frac{dE}{dt} = &\int_{\bar{\Gamma}} \frac{dg(x)}{du_i}\dot{u}_i^0 + \bar{t}_i\dot{w}_i^0\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Gamma^l}(-\lambda^l u^l_{k,k}w^l_{i,i} - \mu^l u^l_{i,j}w^l_{i,j} - \mu^l u^l_{j,i}w^l_{i,j} + (\gamma^l u^l_i - S^l_i)w^l_i)(\dot{\Gamma}^l_m\mathcal{N}^l_m)\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Omega^l} \lambda^l u^l_{k,ki}\dot{w}^l_i + \mu^l u^l_{i,jj}\dot{w}^l_i + \mu^l u^l_{j,ij}\dot{w}^l_i + (\gamma^l u^l_i - S^l_i)\dot{w}^l_i\, d\Omega + ... \\ &+ \sum_{l=0}^{L}\int_{\Omega^l} \lambda^l \dot{u}^l_k w^l_{i,ik} + \mu^l \dot{u}^l_i w^l_{i,jj} + \mu^l \dot{u}^l_j w^l_{i,ji} + \gamma^l \dot{u}^l_i w^l_i\, d\Omega + ... \\ &+ \sum_{l=0}^{L}\int_{\Gamma^l} -\lambda^l u^l_{k,k}\dot{w}^l_i\mathcal{N}^l_i - \mu^l u^l_{i,j}\dot{w}^l_i\mathcal{N}^l_j - \mu^l u^l_{j,i}\dot{w}^l_i\mathcal{N}^l_j\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Gamma^l} -\lambda^l \dot{u}^l_k w^l_{i,i}\mathcal{N}^l_k - \mu^l \dot{u}^l_i w^l_{i,j}\mathcal{N}^l_j - \mu^l \dot{u}^l_j w^l_{i,j}\mathcal{N}^l_i\, d\Gamma. \end{aligned} \tag{305}$$

After re-indexing and combining terms in the last four integrals, the above equation is simplified as

$$\begin{aligned}\frac{dE}{dt} = &\int_{\bar{\Gamma}} \frac{dg(x)}{du_i}\dot{u}_i^0 + \bar{t}_i\dot{w}_i^0\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Gamma^l}(-\lambda^l u^l_{k,k}w^l_{i,i} - \mu^l u^l_{i,j}w^l_{i,j} - \mu^l u^l_{j,i}w^l_{i,j} + (\gamma^l u^l_i - S^l_i)w^l_i)(\dot{\Gamma}^l_m\mathcal{N}^l_m)\, d\Gamma + ... \\ &+ \sum_{l=0}^{L}\int_{\Omega^l} \dot{w}^l_i((\lambda^l + \mu^l)u^l_{k,ki} + \mu^l u^l_{i,kk} + \gamma^l u^l_i - S^l_i)\, d\Omega + ... \end{aligned}$$

$$+\sum_{l=0}^{L}\int_{\Omega^l}\dot{u}_i^l((\lambda^l+\mu^l)w_{k,ki}^l+\mu^l w_{i,kk}^l+\gamma^l w_i^l)\,d\Omega+...$$
$$+\sum_{l=0}^{L}\int_{\Gamma^l}\dot{w}_i^l(-\lambda^l u_{k,k}^l\mathcal{N}_i^l-\mu^l u_{i,j}^l\mathcal{N}_j^l-\mu^l u_{j,i}^l\mathcal{N}_j^l)\,d\Gamma+...$$
$$+\sum_{l=0}^{L}\int_{\Gamma^l}\dot{u}_i^l(-\lambda^l w_{k,k}^l\mathcal{N}_i^l-\mu^l w_{i,j}^l\mathcal{N}_j^l-\mu^l w_{j,i}^l\mathcal{N}_j^l)\,d\Gamma. \tag{306}$$

Observe that the third integral disappears thanks to Eq. (297). If we impose that w_i^l obey the homogeneous PDE

$$(\lambda^l+\mu^l)w_{k,ki}^l+\mu^l w_{i,kk}^l+\gamma^l w_i^l=0\quad\text{in }\Omega^l,\ l=0,...,L, \tag{307}$$

then the fourth integral vanishes as well. The last two integrals are further manipulated by observing that

$$\lambda^l w_{k,k}^l\mathcal{N}_i^l+\mu^l w_{i,j}^l\mathcal{N}_j^l+\mu^l w_{j,i}^l\mathcal{N}_j^l=T_{ij}^{(w^l)}\mathcal{N}_j^l, \tag{308}$$

and similarly for u_i. The second integral can be simplified by the fact that $S_i^l=0$ on the boundaries, because it is an internal source, and by checking that

$$\lambda^l u_{k,k}^l w_{i,i}^l+\mu^l u_{i,j}^l w_{i,j}^l+\mu^l u_{j,i}^l w_{i,j}^l=T_{ij}^{(u^l)}w_{i,j}^l. \tag{309}$$

Thus,

$$\begin{aligned}\frac{dE}{dt}=&\int_{\bar{\Gamma}}\frac{dg(x)}{du_i}\dot{u}_i^0+\bar{t}_i\dot{w}_i^0\,d\Gamma+...\\&+\sum_{l=0}^{L}\int_{\Gamma^l}(-T_{ij}^{(u^l)}w_{i,j}^l+\gamma^l u_i^l w_i^l)(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma+...\\&+\sum_{l=0}^{L}\int_{\Gamma^l}-\dot{w}_i^l T_{ij}^{(u^l)}\mathcal{N}_j^l-\dot{u}_i^l T_{ij}^{(w^l)}\mathcal{N}_j^l\,d\Gamma.\end{aligned} \tag{310}$$

As before, we now extract the integral along Γ_0 from the summations and recall that $\mathcal{N}_i^0=-\mathcal{N}_i^l$, $l=1,...,L$. Also note that the second integral does not produce an integral along the outer boundary $\bar{\Gamma}$ since this boundary is fixed, and so $\dot{\bar{\Gamma}}=0$. We thus have

$$\frac{dE}{dt}=\int_{\bar{\Gamma}}\frac{dg(x)}{du_i}\dot{u}_i^0+\bar{t}_i\dot{w}_i^0-\dot{u}_i^0T_{ij}^{(w^0)}\mathcal{N}_j^0-\dot{w}_i^0T_{ij}^{(u^0)}\mathcal{N}_j^0\,d\Gamma+...$$

$$+\sum_{l=1}^{L}\int_{\Gamma^l}(T_{ij}^{(u^0)}w_{i,j}^0-T_{ij}^{(u^l)}w_{i,j}^l+\gamma^l u_i^l w_i^l-\gamma^0 u_i^0 w_i^0)(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma+...$$
$$+\sum_{l=1}^{L}\int_{\Gamma^l}(\dot{w}_i^0 T_{ij}^{(u^0)}-\dot{w}_i^l T_{ij}^{(u^l)}+\dot{u}_i^0 T_{ij}^{(w^0)}-\dot{u}_i^l T_{ij}^{(w^l)})\mathcal{N}_j^l\,d\Gamma. \tag{311}$$

The first integral vanishes due to Eq. (295b) and if in addition

$$T_{ij}^{(w^0)}\mathcal{N}_j^0=\frac{dg(x)}{du_i}\quad\text{on }\bar{\Gamma} \tag{312}$$

is imposed. In order to combine the second and the third integral, we need to find the relation between the time derivative of the displacements and the time derivative of the boundaries. Taking the total time derivative of Eq. (295c) on Γ^l gives

$$\frac{d}{dt}(u_i^l-u_i^0)=\dot{u}_i^l+u_{i,j}^l\dot{\Gamma}_j^l-\dot{u}_i^0-u_{i,j}^0\dot{\Gamma}_j^l=\dot{u}_i^l-\dot{u}_i^0+(u_{i,j}^l-u_{i,j}^0)\dot{\Gamma}_j^l=0. \tag{313}$$

Note that $\dot{\Gamma}_j^l$ can be expressed in terms of the normal and the tangent vector along Γ^l, so

$$(u_{i,j}^l-u_{i,j}^0)\dot{\Gamma}_j^l=(u_{i,j}^l-u_{i,j}^0)\mathcal{N}_j^l(\dot{\Gamma}_m^l\mathcal{N}_m^l)+(u_{i,j}^l-u_{i,j}^0)\mathcal{T}_j^l(\dot{\Gamma}_m^l\mathcal{T}_m^l). \tag{314}$$

The term $(u_{i,j}^l-u_{i,j}^0)\mathcal{T}_j^l$ is the directional derivative of $u_i^l-u_i^0$ along the tangent of Γ^l and thus vanishes due to Eq. (295c). Substituting back into Eq. (313), we arrive at the relation

$$\dot{u}_i^0=\dot{u}_i^l+(u_{i,j}^l-u_{i,j}^0)\mathcal{N}_j^l(\dot{\Gamma}_m^l\mathcal{N}_m^l). \tag{315}$$

Similarly, using Eq. (302), we have

$$\dot{w}_i^0=\dot{w}_i^l+(w_{i,j}^l-w_{i,j}^0)\mathcal{N}_j^l(\dot{\Gamma}_m^l\mathcal{N}_m^l). \tag{316}$$

Substituting Eq. (315) and Eq. (316) back into Eq. (311), the time derivative of the energy takes the form

$$\frac{dE}{dt}=\sum_{l=1}^{L}\int_{\Gamma^l}(T_{ij}^{(u^0)}w_{i,j}^0-T_{ij}^{(u^l)}w_{i,j}^l+\gamma^l u_i^l w_i^l-\gamma^0 u_i^0 w_i^0)(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma+...$$
$$+\sum_{l=1}^{L}\int_{\Gamma^l}(\dot{w}_i^l T_{ij}^{(u^0)}+(w_{i,k}^l-w_{i,k}^0)\mathcal{N}_k^l(\dot{\Gamma}_m^l\mathcal{N}_m^l)T_{ij}^{(u^0)}-\dot{w}_i^l T_{ij}^{(u^l)})\mathcal{N}_j^l\,d\Gamma+...$$

$$+\sum_{l=1}^{L}\int_{\Gamma^l}(\dot{u}_i^l T_{ij}^{(w^0)}+(u_{i,k}^l-u_{i,k}^0)\mathcal{N}_k^l(\dot{\Gamma}_m^l\mathcal{N}_m^l)T_{ij}^{(w^0)}-\dot{u}_i^l T_{ij}^{(w^l)})\mathcal{N}_j^l\,d\Gamma. \tag{317}$$

Using $\mathcal{N}_i^l\mathcal{N}_j^l=\delta_{ij}$ and rearranging terms we obtain

$$\begin{aligned}\frac{dE}{dt}=&\sum_{l=1}^{L}\int_{\Gamma^l}(T_{ij}^{(u^0)}w_{i,j}^0-T_{ij}^{(u^l)}w_{i,j}^l+\gamma^l u_i^l w_i^l-\gamma^0 u_i^0 w_i^0)(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma+...\\&+\sum_{l=1}^{L}\int_{\Gamma^l}\dot{w}_i^l(T_{ij}^{(u^0)}-T_{ij}^{(u^l)})\mathcal{N}_j^l+(w_{i,j}^l-w_{i,j}^0)T_{ij}^{(u^0)}(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma+...\\&+\sum_{l=1}^{L}\int_{\Gamma^l}\dot{u}_i^l(T_{ij}^{(w^0)}-T_{ij}^{(w^l)})\mathcal{N}_j^l+(u_{i,j}^l-u_{i,j}^0)T_{ij}^{(w^0)}(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma.\end{aligned} \tag{318}$$

The first term in the second integral vanishes thanks to Eq. (295d) and the first term in the last integral vanishes if we impose

$$(T_{ij}^{(w^l)}-T_{ij}^{(w^0)})\mathcal{N}_j^l=0\quad\text{on }\Gamma^l,\ l=1,...,L, \tag{319}$$

leaving us with

$$\begin{aligned}\frac{dE}{dt}=&\sum_{l=1}^{L}\int_{\Gamma^l}(-T_{ij}^{(u^l)}w_{i,j}^l+\gamma^l u_i^l w_i^l-\gamma^0 u_i^0 w_i^0+...\\&\qquad+w_{i,j}^l T_{ij}^{(u^0)}+(u_{i,j}^l-u_{i,j}^0)T_{ij}^{(w^0)})(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma\\=&\sum_{l=1}^{L}\int_{\Gamma^l}((T_{ij}^{(u^0)}-T_{ij}^{(u^l)})w_{i,j}^l+(u_{i,j}^l-u_{i,j}^0)T_{ij}^{(w^0)}+(\gamma^l-\gamma^0)u_i^l w_i^l)(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma\\=&\sum_{l=1}^{L}\int_{\Gamma^l}((T_{ij}^{(w^0)}-T_{ij}^{(w^l)})u_{i,j}^l+(w_{i,j}^l-w_{i,j}^0)T_{ij}^{(u^0)}+(\gamma^l-\gamma^0)u_i^l w_i^l)(\dot{\Gamma}_m^l\mathcal{N}_m^l)\,d\Gamma,\end{aligned} \tag{320}$$

where we used that $u_{i,j}^l T_{ij}^{(w^l)}=w_{i,j}^l T_{ij}^{(u^l)}$ for $l=0,...,L$. Thus, finally, the steepest descent direction for boundary $l=1,...,L$ is given by

$$\dot{\Gamma}_i^l=F^l\mathcal{N}_i^l, \tag{321}$$

where

$$F^l=(T_{ij}^{(u^l)}-T_{ij}^{(u^0)})w_{i,j}^l+(u_{i,j}^0-u_{i,j}^l)T_{ij}^{(w^0)}+(\gamma^0-\gamma^l)u_i^l w_i^l, \tag{322}$$

or equivalently,

$$F^l = (T_{ij}^{(w^l)} - T_{ij}^{(w^0)})u_{i,j}^l + (w_{i,j}^0 - w_{i,j}^l)T_{ij}^{(u^0)} + (\gamma^0 - \gamma^l)u_i^l w_i^l. \tag{323}$$

In these equations, the gradients of the displacement fields can be replaced by their respective strain tensors

$$F^l = (T_{ij}^{(u^l)} - T_{ij}^{(u^0)})\epsilon_{ij}^{(w^l)} + (\epsilon_{ij}^{(u^0)} - \epsilon_{ij}^{(u^l)})T_{ij}^{(w^0)} + (\gamma^0 - \gamma^l)u_i^l w_i^l, \tag{324}$$

or equivalently,

$$F^l = (T_{ij}^{(w^l)} - T_{ij}^{(w^0)})\epsilon_{ij}^{(u^l)} + (\epsilon_{ij}^{(w^0)} - \epsilon_{ij}^{(w^l)})T_{ij}^{(u^0)} + (\gamma^0 - \gamma^l)u_i^l w_i^l, \tag{325}$$

because the stress tensors are symmetric.

APPENDIX C

FUNDAMENTAL SOLUTIONS

C.1 2D Acoustics

We follow the derivation in [53] and fill in some gaps. A function $u^*(x;\xi)$ is called fundamental solution of the Helmholtz equation centered at $\xi \in \mathbb{R}^n$ if it satisfies

$$\Delta u^*(x) + k^2 u^*(x) = -\delta(x-\xi), \quad x \in \mathbb{R}^n, \tag{326}$$

where δ is the Dirac distribution. In two dimensions, switching to polar coordinates where $r = \|x-\xi\|$ and assuming radial symmetry, u^* becomes a function of r only and the equation is written as

$$\frac{1}{r}\left(r u^*_{,r}\right)_{,r} + k^2 u^* = -\frac{\delta(r)}{2\pi r}, \quad r \geq 0, \tag{327}$$

which is an ordinary differential equation in terms of r. The homogeneous solution is given by the Bessel functions of the first kind J_0 and the second kind Y_0 and of order 0 as

$$u^*(r) = C_1 J_0(kr) + C_2 Y_0(kr), \tag{328}$$

where C_1, C_2 are constants to be determined. Integrating the original equation Eq. (326) on a small disk around ξ with radius $\epsilon > 0$

$$\int_{B(\xi,\epsilon)} \Delta u^*(x) + k^2 u^*(x)\, dx = -\int_{B(\xi,\epsilon)} \delta(x-\xi)\, dx = -1 \tag{329}$$

and applying the divergence theorem, we have

$$\int_{\partial B(\xi,\epsilon)} \nabla u^*(x) \cdot \mathcal{N}\, ds + k^2 \int_{B(\xi,\epsilon)} u^*(x)\, dx = -1. \tag{330}$$

Now, using the fact that u^* is radially symmetric and switching to polar coordinates again yields

$$2\pi\epsilon\, u^*_{,r}(\epsilon) + 2\pi k^2 \int_0^\epsilon u^*(r)\, dr = -1. \tag{331}$$

For small r, properties of the Bessel functions are

$$J_0(kr) \to 1, \qquad J_{0,r}(kr) \to 0, \tag{332a}$$

$$Y_0(kr) \to \frac{2}{\pi}\ln(kr) + \text{const}, \qquad Y_{0,r}(kr) \to \frac{2}{\pi}\frac{1}{r}. \tag{332b}$$

Substituting Eq. (328) and Eq. (332) into Eq. (331) while letting $\epsilon \to 0$, we are left with

$$4C_2 + 4C_2k^2 \lim_{\epsilon\to 0}\int_0^\epsilon \ln(kr)\,dr = -1, \tag{333}$$

which implies $C_2 = -1/4$ as the above integral tends to zero. The other constant C_1 is determined by letting $r \to \infty$, in which case the radiation boundary condition [53]

$$u^*_{,r}(r) - iku^*(r) = 0, \quad r \to \infty \tag{334}$$

holds. An expansion at infinity provides the far field behavior

$$J_0(kr) \to \sqrt{\frac{2}{k\pi}}\sqrt{\frac{1}{r}}\left(\cos(kr - \tfrac{\pi}{4}) + \mathcal{O}\left(\tfrac{1}{r}\right)\right), \quad r \to \infty \tag{335a}$$

$$Y_0(kr) \to \sqrt{\frac{2}{k\pi}}\sqrt{\frac{1}{r}}\left(\sin(kr - \tfrac{\pi}{4}) + \mathcal{O}\left(\tfrac{1}{r}\right)\right), \quad r \to \infty \tag{335b}$$

so that the radiation condition Eq. (334) becomes

$$\begin{aligned} 0 = \sqrt{\frac{2}{k\pi}}\sqrt{\frac{1}{r}}\Bigg[& C_1\big(-k\sin(kr - \tfrac{\pi}{4}) - ik\cos(kr - \tfrac{\pi}{4})\big) + ... \\ & - \frac{1}{4}\big(k\cos(kr - \tfrac{\pi}{4}) - ik\sin(kr - \tfrac{\pi}{4})\big)\Bigg]. \end{aligned} \tag{336}$$

Thus, $C_1 = \frac{i}{4}$, and the fundamental solution is determined to be

$$u^*(r) = \frac{i}{4}J_0(kr) - \frac{1}{4}Y_0(kr) = \frac{i}{4}H_0^{(1)}(kr), \tag{337}$$

where $H_0^{(1)}$ is the Hankel function of the first kind and order 0. Back in Cartesian coordinates, the result is

$$u^*(x;\xi) = \frac{i}{4}H_0^{(1)}(k\|x - \xi\|). \tag{338}$$

C.2 2D Elastodynamics

For elastodynamics in two dimensions and in the frequency domain, the fundamental solution $U^*_{ij}(x;\xi)$ is defined as the solution to

$$(\lambda+\mu)U^*_{ik,kj} + \mu U^*_{ij,kk} + \omega^2\rho U^*_{ij} = -\delta(x-\xi)\delta_{ij}. \tag{339}$$

in free space, where $r = \|x-\xi\|$. Note that $\delta(r)$ is the Dirac impulse function and is not to be confused with the Kronecker delta tensor δ_{ij}. The fundamental solution is a tensor $U^*_{ij}(x;\xi) \in \mathbb{C}^{2\times 2}$, whose entries are interpreted as follows: $U^*_{ij}(x;\xi)$ is the jth component of the displacement vector at x caused by a unit norm point load vector located at ξ, which is acting in the ith Euclidean basis direction. In particular, given U^*_{ij} one can compute the displacement vector u^*_j at x, which results from an arbitrary point load vector v_i acting at ξ, as

$$u^*_j(x;\xi) = U^*_{ij}(x;\xi)v_i. \tag{340}$$

Solutions to Eq. (339) are proposed for example in [17, 35, 53, 77], but those solutions do not match. Validation of the correct formula is critical to ensure the correctness of the numerical methods. For this reason, we follow the solution approach in [35] to verify that only [77] provides the correct equation for U^*_{ij}, which is Eq. (3.19) in [77]. Note, however, the typo in [77] Eq. (3.20) where the second ψ should be replaced by χ.

We will first state the fundamental solution and then show some steps of the derivation. First, however, define the following quantities

$$c_p = \sqrt{\frac{\lambda+2\mu}{\rho}}, \quad c_s = \sqrt{\frac{\mu}{\rho}}, \tag{341}$$

$$k_p = \frac{\omega}{c_p}, \qquad k_s = \frac{\omega}{c_s}, \tag{342}$$

where c_p is the compressional and c_s the shear wave speed, and k_p and k_s are the corresponding wave numbers. Then, the fundamental solution is given by

$$U^*_{ij} = \frac{1}{2\pi\mu}(\psi\delta_{ij} - \chi r_{,i}r_{,j}), \tag{343}$$

where

$$\psi = K_0(ik_s r) + \frac{K_1(ik_s r)}{ik_s r} - \frac{k_p^2}{k_s^2}\frac{K_1(ik_p r)}{ik_p r}, \tag{344}$$

$$\chi = K_2(ik_s r) - \frac{k_p^2}{k_s^2} K_2(ik_p r), \tag{345}$$

and $K_\alpha(\cdot)$ is the modified Bessel function of the second kind and order α.

In order to corroborate this result, some calculations from the derivation in [35] are shown in detail. For a higher-level insight and discussion, the reader is referred to [35]. Consider, for example, the first row of U_{ij}^*, i.e., $u_j^* = U_{1j}^*$. Then, using vector notation for a moment,

$$\boldsymbol{u}^* = \nabla \cdot (\nabla \boldsymbol{A}^p) - \nabla \times \nabla \times \boldsymbol{A}^s, \tag{346}$$

where the potentials $\boldsymbol{A}^p$ and $\boldsymbol{A}^s$ are written as $\boldsymbol{A}^p = A^p \boldsymbol{e_1}$ and $\boldsymbol{A}^s = A^s \boldsymbol{e_1}$, where $\boldsymbol{e_1}$ is the first base vector, or equivalently in index notation

$$u_i^* = A^p_{j,ij} - (A^s_{j,ij} - A^s_{i,jj}). \tag{347}$$

Thus, in two dimensions we obtain for the components

$$u_1^* = A^p_{1,11} + A^p_{2,12} - A^s_{1,11} - A^s_{2,12} + A^s_{1,11} + A^s_{1,22} = A^p_{1,11} + A^s_{1,22} = A^p_{,11} + A^s_{,22}, \tag{348a}$$

$$u_2^* = A^p_{1,21} + A^p_{2,22} - A^s_{1,21} - A^s_{2,22} + A^s_{2,11} + A^s_{2,22} = A^p_{1,21} - A^s_{1,21} = A^p_{,21} - A^s_{,21}. \tag{348b}$$

Now, write the potentials as

$$A^p = C\ln(r) + C_{1p} K_0(ik_p r) + C_{2p} I_0(ik_p r), \tag{349}$$

$$A^s = C\ln(r) + C_{1s} K_0(ik_s r) + C_{2s} I_0(ik_s r), \tag{350}$$

where $I_0(\cdot)$ is the modified Bessel function of the first kind and order 0. Equilibrium and radiation conditions yield $C_{2p} = C_{2s} = 0$ and $C_{1p} = C_{1s} = -1/(2\pi\mu k_s^2) =$

$-1/(2\pi\rho\omega^2)$. In the following, we make use of the identities for modified Bessel functions

$$\frac{d}{dr}K_0(ikr) = -ikK_1(ikr), \tag{351a}$$

$$\frac{d}{dr}K_1(ikr) = -\frac{ik}{2}(K_0(ikr) + K_2(ikr)), \tag{351b}$$

and the derivatives of the radius function $r = \|x - \xi\|$

$$r_{,1} = \frac{d_1}{r}, \quad r_{,2} = \frac{d_2}{r}, \tag{352a}$$

$$r_{,11} = \frac{d_2^2}{r^3}, \quad r_{,22} = \frac{d_1^2}{r^3}, \quad r_{,12} = -\frac{d_1 d_2}{r^3}, \tag{352b}$$

where d_1 (d_2) is the first (second) component of the vector $d_i = x - \xi$, i.e., $r = \sqrt{d_1^2 + d_2^2}$. Carrying out the derivatives in Eq. (348a), one observes that the constant C cancels and one arrives at

$$\begin{aligned} 2\pi\mu u_1^* &= \frac{1}{2}\frac{k_p^2}{k_s^2}K_0(ik_p r)\frac{d_1^2}{r^2} + \frac{1}{2}K_0(ik_s r)\frac{d_2^2}{r^2} + \frac{1}{2}\frac{k_p^2}{k_s^2}K_2(ik_p r)\frac{d_1^2}{r^2} + \frac{1}{2}K_2(ik_s r)\frac{d_2^2}{r^2} + ... \\ &\quad - \frac{k_p}{ik_s^2}K_1(ik_p r)\frac{d_2^2}{r^3} - \frac{1}{ik_s}K_1(ik_s r)\frac{d_1^2}{r^3} = ... \end{aligned} \tag{353a}$$

$$\begin{aligned} &= \frac{1}{2}\frac{k_p^2}{k_s^2}\left(K_2(ik_p r) - \frac{2}{ik_p r}K_1(ik_p r)\right)\frac{d_1^2}{r^2} + \frac{1}{2}K_0(ik_s r)\frac{d_2^2}{r^2} + \frac{1}{2}\frac{k_p^2}{k_s^2}K_2(ik_p r)\frac{d_1^2}{r^2} + ... \\ &\quad + \frac{1}{2}K_2(ik_s r)\frac{d_2^2}{r^2} - \frac{k_p}{ik_s^2}K_1(ik_p r)\frac{d_2^2}{r^3} + \frac{1}{2}\left(K_0(ik_s r) - K_2(ik_s r)\right)\frac{d_1^2}{r^2}, \end{aligned} \tag{353b}$$

where in the first and the last term the identity

$$K_1(ikr) = \frac{ikr}{2}(K_2(ikr) - K_0(ikr)) \tag{354}$$

was employed. Combining terms in Eq. (353b) yields

$$2\pi\mu u_1^* = \frac{k_p^2}{k_s^2}K_2(ik_p r)\frac{d_1^2}{r^2} - \frac{k_p^2}{k_s^2}\frac{K_1(ik_p r)}{ik_p r} + \frac{1}{2}K_0(ik_s r) + \frac{1}{2}K_2(ik_s r)\frac{d_2^2 - d_1^2}{r^2}. \tag{355}$$

Using Eq. (354), the last term can be simplified as

$$\frac{1}{2}K_2(ik_s r)\frac{d_2^2 - d_1^2}{r^2} = \frac{1}{2}K_2(ik_s r)\frac{r^2 - 2d_1^2}{r^2} = \frac{1}{2}K_2(ik_s r) - K_2(ik_s r)\frac{d_1^2}{r^2} = ...$$

$$= \frac{1}{2}\left(K_0(ik_s r) + \frac{2}{ik_s r}K_1(ik_s r)\right) - K_2(ik_s r)\frac{d_1^2}{r^2} = ...$$
$$= \frac{1}{2}K_0(ik_s r) + \frac{K_1(ik_s r)}{ik_s r} - K_2(ik_s r)\frac{d_1^2}{r^2}. \tag{356}$$

Thus, Eq. (353b) becomes

$$2\pi\mu u_1^* = K_0(ik_s r) + \frac{K_1(ik_s r)}{ik_s r} - \frac{k_p^2}{k_s^2}\frac{K_1(ik_p r)}{ik_p r} + \left(\frac{k_p^2}{k_s^2}K_2(ik_p r) - K_2(ik_s r)\right)\frac{d_1^2}{r^2} = ...$$
$$= \psi - \chi r_{,1} r_{,1}, \tag{357}$$

which is in accordance with Eq. (343). Similar computations are carried out to obtain u_2^* from Eq. (348b). After differentiation,

$$2\pi\mu u_2^* = \left(\frac{1}{2}\frac{k_p^2}{k_s^2}K_0(ik_p r) - \frac{1}{2}K_0(ik_s r) + \frac{1}{2}\frac{k_p^2}{k_s^2}K_2(ik_p r) - \frac{1}{2}K_2(ik_s r)\right)\frac{d_1 d_2}{r^2} + ...$$
$$+ \left(\frac{k_p}{ik_s^2}K_1(ik_p r) - \frac{1}{ik_s}K_1(ik_s r)\right)\frac{d_1 d_2}{r^3} \tag{358a}$$
$$= \left(\frac{1}{2}\frac{k_p^2}{k_s^2}K_0(ik_p r) - \frac{1}{2}K_0(ik_s r) + \frac{1}{2}\frac{k_p^2}{k_s^2}K_2(ik_p r) - \frac{1}{2}K_2(ik_s r)\right)\frac{d_1 d_2}{r^2} + ...$$
$$+ \left(-\frac{1}{2}\frac{k_p^2}{k_s^2}K_0(ik_p r) + \frac{1}{2}K_0(ik_s r) + \frac{1}{2}\frac{k_p^2}{k_s^2}K_2(ik_p r) - \frac{1}{2}K_2(ik_s r)\right)\frac{d_1 d_2}{r^2} \tag{358b}$$
$$= \left(\frac{k_p^2}{k_s^2}K_2(ik_p r) - K_2(ik_s r)\right)\frac{d_1 d_2}{r^2} = -\chi r_{,1} r_{,2}, \tag{358c}$$

where again Eq. (354) was employed to obtain Eq. (358c). Thus, the fundamental solution in Eq. (343) is correct for U_{12}^* as well. Validation of Eq. (343) for the second row of U_{ij}^* is done analogously by swapping the 1 and 2 direction.

It remains to provide the equations for the strain, stress and traction induced by the fundamental displacement solution. First, differentiation with respect to the coordinate axes yields

$$U_{ij,k}^* = \frac{1}{2\pi\mu}(\psi_{,k}\delta_{ij} - \chi_{,k} r_{,i} r_{,j} - \chi r_{,ik} r_{,j} - \chi r_{,i} r_{,jk})$$
$$= \frac{1}{2\pi\mu}\left(\psi_{,r} r_{,k}\delta_{ij} - \chi_{,r} r_{,i} r_{,j} r_{,k} - \frac{\chi}{r}(r_{,j}\delta_{ik} + r_{,i}\delta_{jk} - 2r_{,i} r_{,j} r_{,k})\right), \tag{359}$$

where we used that $\chi_{,k} = \chi_{,r} r_{,k}$ and $\psi_{,k} = \psi_{,r} r_{,k}$ by the chain rule and

$$r_{,ij} = \frac{\delta_{ij} - r_{,i} r_{,j}}{r}. \tag{360}$$

From Eq. (95), the fundamental strain tensor

$$\begin{aligned} \epsilon^*_{ijk} = \frac{1}{4\pi\mu}\Bigg(& \psi_{,r}(r_{,k}\delta_{ij} + r_{,j}\delta_{ik}) - 2\chi_{,r} r_{,i} r_{,j} r_{,k} + ... \\ & - \frac{\chi}{r}(r_{,j}\delta_{ik} + r_{,k}\delta_{ij} + 2r_{,i}\delta_{jk} - 4r_{,i} r_{,j} r_{,k})\Bigg) \end{aligned} \tag{361}$$

follows using Eq. (359). Now, using Eq. (97), the fundamental stress tensor

$$\begin{aligned} T^*_{ijk} = \frac{1}{2\pi}\Bigg[& \left(\tfrac{k_s^2}{k_p^2} - 2\right)\left(\psi_{,r} - \chi_{,r} - \frac{\chi}{r}\right) r_{,i}\delta_{jk} + ... \\ & + \psi_{,r}(r_{,k}\delta_{ij} + r_{,j}\delta_{ik})\Bigg) - 2\chi_{,r} r_{,i} r_{,j} r_{,k} + ... \\ & - \frac{\chi}{r}(r_{,j}\delta_{ik} + r_{,k}\delta_{ij} + 2r_{,i}\delta_{jk} - 4r_{,i} r_{,j} r_{,k})\Bigg] \end{aligned} \tag{362}$$

is obtained, where we made use of the relationships $r_{,k} r_{,k} = 1$, $\delta_{kk} = 2$ and

$$\frac{k_s^2}{k_p^2} - 2 = \frac{c_p^2}{c_s^2} - 2 = \frac{\lambda + 2\mu}{\mu} - 2 = \frac{\lambda}{\mu}. \tag{363}$$

From Eq. (96), one obtains the fundamental traction tensor on a plane with normal direction $\mathcal{N}_i$ as

$$\begin{aligned} t^*_{ij} = \frac{1}{2\pi}\Bigg[& \left(\tfrac{k_s^2}{k_p^2} - 2\right)\left(\psi_{,r} - \chi_{,r} - \frac{\chi}{r}\right) r_{,i}\,\mathcal{N}_j + ... \\ & + \psi_{,r}\left(\delta_{ij} r_{,k}\,\mathcal{N}_k + r_{,j}\,\mathcal{N}_i\right) - 2\chi_{,r} r_{,i} r_{,j} r_{,k}\,\mathcal{N}_k + ... \\ & - \frac{\chi}{r}(r_{,j}\,\mathcal{N}_i + \delta_{ij} r_{,k}\,\mathcal{N}_k + 2r_{,i}\,\mathcal{N}_j - 4r_{,i} r_{,j} r_{,k}\,\mathcal{N}_k)\Bigg], \end{aligned} \tag{364}$$

where $r_{,k}\,\mathcal{N}_k = \partial r/\partial\mathcal{N} = \nabla r \cdot \mathcal{N} = (d_1\mathcal{N}_1 + d_2\mathcal{N}_2)/r$ is the directional derivative of r in the direction $\mathcal{N}_i$. Analogous to the fundamental displacement tensor, the traction vector resulting from a point load vector v_i located at ξ can be obtained by

$$t^*_j(x;\xi) = t^*_{ij}(x;\xi) v_i. \tag{365}$$

For convenience when implementing these equations, we write out and simplify the components of the displacement and traction tensors as follows:

$$U^*_{11} = \frac{\psi - \chi\, r_{,1}^2}{2\pi\mu}, \tag{366a}$$

$$U^*_{12} = -\frac{\chi\, r_{,1} r_{,2}}{2\pi\mu} = U^*_{21}, \tag{366b}$$

$$U^*_{22} = \frac{\psi - \chi\, r_{,2}^2}{2\pi\mu}, \tag{366c}$$

and

$$\begin{aligned} t^*_{11} = \frac{1}{2\pi}\Bigg[&\left(\tfrac{k_s^2}{k_p^2} - 2\right)\left(\psi_{,r} - \chi_{,r} - \frac{\chi}{r}\right) r_{,1}\,\mathcal{N}_1 + ... \\ &+ \psi_{,r}\left(r_{,1}\,\mathcal{N}_1 + \frac{\partial r}{\partial \mathcal{N}}\right) - 2\chi_{,r} r_{,1}^2 \frac{\partial r}{\partial \mathcal{N}} - \frac{\chi}{r}\left(3 r_{,1}\,\mathcal{N}_1 + (1 - 4 r_{,1}^2)\frac{\partial r}{\partial \mathcal{N}}\right)\Bigg], \end{aligned} \tag{367a}$$

$$\begin{aligned} t^*_{12} = \frac{1}{2\pi}\Bigg[&\left(\tfrac{k_s^2}{k_p^2} - 2\right)\left(\psi_{,r} - \chi_{,r} - \frac{\chi}{r}\right) r_{,1}\,\mathcal{N}_2 + ... \\ &+ \psi_{,r} r_{,2}\,\mathcal{N}_1 - 2\chi_{,r} r_{,1} r_{,2} \frac{\partial r}{\partial \mathcal{N}} - \frac{\chi}{r}\left(r_{,2}\,\mathcal{N}_1 + 2 r_{,1}\,\mathcal{N}_2 - 4 r_{,1} r_{,2}\frac{\partial r}{\partial \mathcal{N}}\right)\Bigg], \end{aligned} \tag{367b}$$

$$\begin{aligned} t^*_{21} = \frac{1}{2\pi}\Bigg[&\left(\tfrac{k_s^2}{k_p^2} - 2\right)\left(\psi_{,r} - \chi_{,r} - \frac{\chi}{r}\right) r_{,2}\,\mathcal{N}_1 + ... \\ &+ \psi_{,r} r_{,1}\,\mathcal{N}_2 - 2\chi_{,r} r_{,1} r_{,2} \frac{\partial r}{\partial \mathcal{N}} - \frac{\chi}{r}\left(r_{,1}\,\mathcal{N}_2 + 2 r_{,2}\,\mathcal{N}_1 - 4 r_{,1} r_{,2}\frac{\partial r}{\partial \mathcal{N}}\right)\Bigg], \end{aligned} \tag{367c}$$

$$\begin{aligned} t^*_{22} = \frac{1}{2\pi}\Bigg[&\left(\tfrac{k_s^2}{k_p^2} - 2\right)\left(\psi_{,r} - \chi_{,r} - \frac{\chi}{r}\right) r_{,2}\,\mathcal{N}_2 + ... \\ &+ \psi_{,r}\left(r_{,2}\,\mathcal{N}_2 + \frac{\partial r}{\partial \mathcal{N}}\right) - 2\chi_{,r} r_{,2}^2 \frac{\partial r}{\partial \mathcal{N}} - \frac{\chi}{r},\left(3 r_{,2}\,\mathcal{N}_2 + (1 - 4 r_{,2}^2)\frac{\partial r}{\partial \mathcal{N}}\right)\Bigg], \end{aligned} \tag{367d}$$

where, as a reminder,

$$\psi = K_0(ik_s r) + \frac{K_1(ik_s r)}{ik_s r} - \frac{k_p^2}{k_s^2}\frac{K_1(ik_p r)}{ik_p r}, \tag{368a}$$

$$\chi = K_2(ik_s r) - \frac{k_p^2}{k_s^2} K_2(ik_p r), \tag{368b}$$

and the derivatives with respect to r are obtained using the identities

$$\frac{dK_0(ik_s r)}{dr} = -ik_s K_1(ik_s r), \tag{369a}$$

$$\frac{dK_1(ik_s r)}{dr} = -\frac{ik_s}{2}\left(K_0(ik_s r) + K_0(ik_s r)\right), \tag{369b}$$

$$\frac{dK_1(ik_pr)}{dr} = -\frac{ik_p}{2}\left(K_0(ik_pr) + K_0(ik_pr)\right), \tag{369c}$$

$$\frac{dK_2(ik_sr)}{dr} = -\frac{ik_s}{2}\left(K_1(ik_sr) + K_3(ik_sr)\right), \tag{369d}$$

$$\frac{dK_2(ik_pr)}{dr} = -\frac{ik_p}{2}\left(K_1(ik_pr) + K_3(ik_pr)\right). \tag{369e}$$

REFERENCES

[1] Babuska, I. and Sauter, S., "Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers?," *SIAM review*, vol. 42, no. 3, pp. 451–484, 2000.

[2] Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M., and Szeliski, R., "A database and evaluation methodology for optical flow," *International Journal of Computer Vision*, pp. 1–31, 2007.

[3] Banerjee, P. and Butterfield, R., "Boundary element methods in geomechanics," *Finite elements in geomechanics*, pp. 529–570, 1977.

[4] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L., "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[5] Benamou, J. and Brenier, Y., "A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem," *Numerische Mathematik*, vol. 84, no. 3, pp. 375–393, 2000.

[6] Benedetti, M., Lesselier, D., Lambert, M., and Massa, A., "Multiple-shape reconstruction by means of multiregion level sets," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 5, pp. 2330–2342, 2010.

[7] Béréziat, D., Herlin, I., and Younes, L., "A generalized optical flow constraint and its physical interpretation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 487–492, IEEE, 2000.

[8] Bignardi, S., Santarato, G., and Fedele, F., *Complete waveform inversion approach to seismic surface waves and adjoint active surfaces*. PhD thesis, Universita degli Studi di Ferrara, 2010.

[9] Bishop, C., *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[10] Black, M. and Anandan, P., "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Computer Vision and Image Understanding*, vol. 63, no. 1, pp. 75–104, 1996.

[11] Bonnet, M., "BIE and material differentiation applied to the formulation of obstacle inverse problems," *Engineering analysis with boundary elements*, vol. 15, no. 2, pp. 121–136, 1995.

[12] Bonnet, M. and Guzina, B., "Sounding of finite solid bodies by way of topological derivative," *International journal for numerical methods in engineering*, vol. 61, no. 13, pp. 2344–2373, 2004.

[13] Bonnet, M. and Guzina, B., "Elastic-wave identification of penetrable obstacles using shape-material sensitivity framework," *Journal of Computational Physics*, vol. 228, no. 2, pp. 294–311, 2009.

[14] Borges, P. and Izquierdo, E., "A probabilistic approach for vision-based fire detection in videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 5, pp. 721–731, 2010.

[15] Botev, Z., Grotowski, J., and Kroese, D., "Kernel density estimation via diffusion," *The Annals of Statistics*, vol. 38, no. 5, pp. 2916–2957, 2010.

[16] Brebbia, C. and Dominguez, J., "Boundary element methods for potential problems," *Applied Mathematical Modelling*, vol. 1, no. 7, pp. 372–378, 1977.

[17] Brebbia, C. and Dominguez, J., *Boundary Elements: An Introductory Course*. WIT Press (UK), 1992.

[18] Bresson, X., Vandergheynst, P., and Thiran, J.-P., "A variational model for object segmentation using boundary information and shape prior driven by the Mumford-Shah functional," *International Journal of Computer Vision*, vol. 68, no. 2, pp. 145–162, 2006.

[19] Brox, T. and Malik, J., "Object segmentation by long term analysis of point trajectories," in *European Conference on Computer Vision (ECCV)*, pp. 282–295, Springer, 2010.

[20] Burger, M. and Osher, S., "A survey on level set methods for inverse problems and optimal design," *European Journal of Applied Mathematics*, vol. 16, no. 2, pp. 263–301, 2005.

[21] Cakoni, F. and Colton, D., *Qualitative Methods in Inverse Scattering Theory: An Introduction*. Springer, 2005.

[22] Çelik, T. and Demirel, H., "Fire detection in video sequences using a generic color model," *Fire Safety Journal*, vol. 44, no. 2, pp. 147–158, 2009.

[23] Chan, T. and Vese, L., "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[24] Chen, Y., Thiruvenkadam, S., Tagare, H., Huang, F., Wilson, D., and Geiser, E., "On the incorporation of shape priors into geometric active contours," in *In Proceedings of IEEE Workshop on Variational and Level Set Methods in Computer Vision*, pp. 145–152, IEEE, 2001.

[25] Cheney, M., "The linear sampling method and the MUSIC algorithm," *Inverse Problems*, vol. 17, no. 4, p. 591, 2001.

[26] Colton, D., Coyle, J., and Monk, P., "Recent developments in inverse acoustic scattering theory," *Siam Review*, vol. 42, no. 3, pp. 369–414, 2000.

[27] Colton, D., Haddar, H., and Piana, M., "The linear sampling method in inverse electromagnetic scattering theory," *Inverse problems*, vol. 19, no. 6, p. S105, 2003.

[28] Colton, D. and Kress, R., *Inverse Acoustic and Electromagnetic Scattering Theory*, vol. 93. Springer, 1998.

[29] Cook, D., Fedele, F., and Yezzi, A., "Detection of spherical inclusions using active surfaces," in *Proceedings of the Institute of Acoustics, International Conference on Synthetic Aperture Sonar and Synthetic Aperture Radar; Lerici, Italy*, 2010.

[30] Cootes, T., Edwards, G., and Taylor, C., "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.

[31] Cootes, T., Taylor, C., Cooper, D., and Graham, J., "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[32] Cremers, D., Rousson, M., and Deriche, R., "A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 195–215, 2007.

[33] Do Carmo, M., *Differential Geometry of Curves and Surfaces*, vol. 2. Prentice-Hall Englewood Cliffs, 1976.

[34] Dominguez, J., *Boundary Elements in Dynamics*. Computational Mechanics Publications, Southampton, 1993.

[35] Domínguez, J. and Abascal, R., "On fundamental solutions for the boundary integral equations method in static and dynamic elasticity," *Engineering Analysis*, vol. 1, no. 3, pp. 128–134, 1984.

[36] Doretto, G., "Modeling dynamic scenes with active appearance," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 66–73, IEEE, 2005.

[37] Doretto, G. and Soatto, S., "Dynamic shape and appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2006–2019, 2006.

[38] Dorn, O. and Lesselier, D., "Level set methods for inverse scattering – some recent developments," *Inverse Problems*, vol. 25, no. 12, p. 125001, 2009.

[39] Dorn, O., Miller, E., and Rappaport, C., "A shape reconstruction method for electromagnetic tomography using adjoint fields and level sets," *Inverse problems*, vol. 16, no. 5, p. 1119, 2000.

[40] Du Bartas, G., *La Sepmaine ou Creation du Monde*. Michel Gadoulleau et Jean Febvrier, Paris, 1578.

[41] Eskandari, M. and Safian, R., "Inverse scattering method based on contour deformations using a fast marching method," *Inverse Problems*, vol. 26, no. 9, p. 095002, 2010.

[42] Eymard, R., Gallouët, T., and Herbin, R., "Finite volume methods," *Handbook of numerical analysis*, vol. 7, pp. 713–1018, 2000.

[43] Fairweather, G. and Karageorghis, A., "The method of fundamental solutions for elliptic boundary value problems," *Advances in Computational Mathematics*, vol. 9, no. 1, pp. 69–95, 1998.

[44] Fata, S., Guzina, B., and Bonnet, M., "Computational framework for the BIE solution to inverse scattering problems in elastodynamics," *Computational mechanics*, vol. 32, no. 4, pp. 370–380, 2003.

[45] Fazekas, S., Amiaz, T., Chetverikov, D., and Kiryati, N., "Dynamic texture detection based on motion analysis," *International Journal of Computer Vision*, vol. 82, no. 1, pp. 48–63, 2009.

[46] Fazekas, S. and Chetverikov, D., "Analysis and performance evaluation of optical flow features for dynamic texture recognition," *Signal Processing: Image Communication*, vol. 22, no. 7, pp. 680–691, 2007.

[47] Feijóo, G., "A new method in inverse scattering based on the topological derivative," *Inverse Problems*, vol. 20, no. 6, p. 1819, 2004.

[48] Feijóo, G., Oberai, A., and Pinsky, P., "An application of shape optimization in the solution of inverse acoustic scattering problems," *Inverse problems*, vol. 20, no. 1, p. 199, 2003.

[49] Fink, M., Cassereau, D., Derode, A., Prada, C., Roux, P., Tanter, M., Thomas, J., and Wu, F., "Time-reversed acoustics," *Reports on Progress in Physics*, vol. 63, no. 12, p. 1933, 2000.

[50] Foulonneau, A., Charbonnier, P., and Heitz, F., "Affine-invariant geometric shape priors for region-based active contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1352–1357, 2006.

[51] Fritscher, K., Grünerbl, A., and Schubert, R., "3D image segmentation using combined shape-intensity prior models," *International Journal of Computer Assisted Radiology and Surgery*, vol. 1, no. 6, pp. 341–350, 2007.

[52] Gangbo, W. and McCann, R., "The geometry of optimal transportation," *Acta Mathematica*, vol. 177, no. 2, pp. 113–161, 1996.

[53] Gaul, L., Kögl, M., and Wagner, M., *Boundary Element Methods for Engineers and Scientists*. Springer, 2003.

[54] Gelfand, I. and Fomin, S., *Calculus of Variations*. Courier Dover Publications, 2000.

[55] George, P., *Automatic Mesh Generation: Applications to Finite Element Methods*. John Wiley & Sons, Inc., 1992.

[56] Graff, K., *Wave Motion in Elastic Solids*. Dover publications, 1991.

[57] Gruber, F., Marengo, E., and Devaney, A., "Time-reversal imaging with multiple signal classification considering multiple scattering between the targets," *The Journal of the Acoustical Society of America*, vol. 115, p. 3042, 2004.

[58] Guzina, B. and Bonnet, M., "Topological derivative for the inverse scattering of elastic waves," *The Quarterly Journal of Mechanics and Applied Mathematics*, vol. 57, no. 2, pp. 161–179, 2004.

[59] Guzina, B., Nintcheu Fata, S., and Bonnet, M., "On the stress-wave imaging of cavities in a semi-infinite solid," *International journal of solids and structures*, vol. 40, no. 6, pp. 1505–1523, 2003.

[60] Haber, E., Rehman, T., and Tannenbaum, A., "An efficient numerical method for the solution of the L2 optimal mass transfer problem," *SIAM Journal on Scientific Computing*, vol. 32, no. 1, pp. 197–211, 2010.

[61] Habiboğlu, Y., Günay, O., and Çetin, A., "Covariance matrix-based fire and flame detection method in video," *Machine Vision and Applications*, vol. 23, no. 6, pp. 1103–1113, 2012.

[62] Hajihashemi, M. and El-Shenawee, M., "Level set algorithm for shape reconstruction of non-overlapping three-dimensional penetrable targets," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 1, pp. 75–86, 2012.

[63] Haker, S., Zhu, L., Tannenbaum, A., and Angenent, S., "Optimal mass transport for registration and warping," *International Journal of Computer Vision*, vol. 60, no. 3, pp. 225–240, 2004.

[64] He, L., Kindermann, S., and Sini, M., "Reconstruction of shapes and impedance functions using few far-field measurements," *Journal of Computational Physics*, vol. 228, no. 3, pp. 717–730, 2009.

[65] Heimann, T. and Meinzer, H.-P., "Statistical shape models for 3D medical image segmentation: A review," *Medical Image Analysis*, vol. 13, no. 4, pp. 543–563, 2009.

[66] Herrera, I., *Boundary methods: an algebraic theory*. Pitman Advanced Pub. Program, 1984.

[67] Ho, C., "Machine vision-based real-time early flame and smoke detection," *Measurement Science and Technology*, vol. 20, p. 045502, 2009.

[68] Horn, B. and Schunck, B., "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[69] Hu, S. and Collins, D., "Joint level-set shape modeling and appearance modeling for brain structure segmentation," *NeuroImage*, vol. 36, no. 3, pp. 672–683, 2007.

[70] Huang, X. and Metaxas, D., "Metamorphs: deformable shape and appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1444–1459, 2008.

[71] Jackson, J., *Layered Deformotion with Radiance: A Model for Appearance, Segmentation, Registration, and Tracking*. PhD thesis, Georgia Institute of Technology, 2007.

[72] Jackson, J., Yezzi, A., and Soatto, S., "Joint priors for variational shape and appearance modeling," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[73] Jain, A., Murty, M., and Flynn, P., "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[74] Kallivokas, L., Fathi, A., Kucukcoban, S., Stokoe II, K., Bielak, J., and Ghattas, O., "Site characterization using full waveform inversion," *Soil Dynamics and Earthquake Engineering*, 2013.

[75] Kantorovich, L., "On the translocation of masses," *Journal of Mathematical Sciences*, vol. 133, no. 4, pp. 1381–1382, 2006.

[76] Kass, M., Witkin, A., and Terzopoulos, D., "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[77] Kausel, E., *Fundamental Solutions in Elastodynamics: A Compendium*. Cambridge University Press, 2006.

[78] Kichenassamy, S., Kumar, A., Olver, P., Tannenbaum, A., and Yezzi, A., "Gradient flows and geometric active contour models," in *International Conference on Computer Vision (ICCV)*, pp. 810–815, IEEE, 1995.

[79] Kim, J., Fisher III, J., Yezzi, A., Çetin, M., and Willsky, A., "A non-parametric statistical method for image segmentation using information theory and curve evolution," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1486–1502, 2005.

[80] Kirsch, A., "Characterization of the shape of a scattering obstacle using the spectral data of the far field operator," *Inverse Problems*, vol. 14, no. 6, p. 1489, 1998.

[81] Kirsch, A., *An Introduction to the Mathematical Theory of Inverse Problems*, vol. 120. Springer, 2011.

[82] Kita, E. and Kamiya, N., "Trefftz method: an overview," *Advances in Engineering Software*, vol. 24, no. 1, pp. 3–12, 1995.

[83] Kitagawa, T., "On the numerical stability of the method of fundamental solution applied to the Dirichlet problem," *Japan Journal of Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 123–133, 1988.

[84] Kitagawa, T., "Asymptotic stability of the fundamental solution method," *Journal of Computational and Applied Mathematics*, vol. 38, no. 1, pp. 263–269, 1991.

[85] Kleinman, R. and van den Berg, P., "A modified gradient method for two-dimensional problems in tomography," *Journal of Computational and Applied Mathematics*, vol. 42, no. 1, pp. 17–35, 1992.

[86] Ko, B., Cheong, K., and Nam, J., "Fire detection based on vision sensor and support vector machines," *Fire Safety Journal*, vol. 44, no. 3, pp. 322–329, 2009.

[87] Kolesov, I., Karasev, P., Tannenbaum, A., and Haber, E., "Fire and smoke detection in video with optimal mass transport based optical flow and neural networks," in *IEEE International Conference on Image Processing (ICIP)*, pp. 761–764, IEEE, 2010.

[88] Leventon, M., Grimson, W., and Faugeras, O., "Statistical shape influence in geodesic active contours," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 316–323, IEEE, 2000.

[89] Li, Z.-C., Lu, T.-T., Hu, H.-Y., and Cheng, A., *Trefftz and Collocation Methods*. WIT Press, Southampton, 2008.

[90] Litman, A., Lesselier, D., and Santosa, F., "Reconstruction of a two-dimensional binary obstacle by controlled evolution of a level-set," *Inverse Problems*, vol. 14, no. 3, p. 685, 1999.

[91] Liu, C. and Ahuja, N., "Vision based fire detection," in *International Conference on Pattern Recognition (ICPR)*, vol. 4, pp. 134–137, IEEE, 2004.

[92] Lowe, D., "Object recognition from local scale-invariant features," in *International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1150–1157, IEEE, 1999.

[93] Lowe, D., "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[94] Lucas, B. and Kanade, T., "An iterative image registration technique with an application to stereo vision," in *International Joint Conference on Artificial Intelligence*, vol. 2, pp. 674–679, IJCAI, 1981.

[95] Malvern, L., *Introduction to the Mechanics of a Continuous Medium*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1969.

[96] Marbach, G., Loepfe, M., and Brupbacher, T., "An image processing technique for fire detection in video images," *Fire Safety Journal*, vol. 41, no. 4, pp. 285–289, 2006.

[97] Marengo, E., Gruber, F., and Simonetti, F., "Time-reversal MUSIC imaging of extended targets," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 1967–1984, 2007.

[98] Michailovich, O., Rathi, Y., and Tannenbaum, A., "Image segmentation using active contours driven by the Bhattacharyya gradient flow," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2787–2801, 2007.

[99] Miezianko, R., "Terravic research infrared database," *IEEE OTCBVS WS Series Bench*.

[100] Morris, N., Avidan, S., Matusik, W., and Pfister, H., "Statistics of infrared images," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–7, IEEE, 2007.

[101] Mueller, M., Karasev, P., Kolesov, I., and Tannenbaum, A., "A video analytics framework for amorphous and unstructured anomaly detection," in *Proceedings of International Conference on Image Processing (ICIP)*, IEEE, 2011.

[102] Mueller, M., Karasev, P., Kolesov, I., and Tannenbaum, A., "Optical flow estimation for flame detection in videos," *IEEE Transactions on Image Processing*, 2013.

[103] Mumford, D. and Shah, J., "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[104] Nabavi, M., Siddiqui, M., and Dargahi, J., "A new 9-point sixth-order accurate compact finite-difference method for the Helmholtz equation," *Journal of Sound and Vibration*, vol. 307, no. 3, pp. 972–982, 2007.

[105] Novotny, A., Feijóo, R., Taroco, E., and Padra, C., "Topological sensitivity analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 192, no. 7, pp. 803–829, 2003.

[106] Osher, S. and Sethian, J., "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation," *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988.

[107] Papenberg, N., Bruhn, A., Brox, T., Didas, S., and Weickert, J., "Highly accurate optic flow computation with theoretically justified warping," *International Journal of Computer Vision*, vol. 67, no. 2, pp. 141–158, 2006.

[108] Perona, P. and Malik, J., "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[109] Phillips III, W., Shah, M., and da Vitoria Lobo, N., "Flame recognition in video," *Pattern Recognition Letters*, vol. 23, no. 1-3, pp. 319–327, 2002.

[110] Rousson, M. and Paragios, N., "Shape priors for level set representations," in *European Conference on Computer Vision (ECCV)*, pp. 78–92, Springer, 2002.

[111] Rus, G. and Gallego, R., "Solution of identification inverse problems in elastodynamics using semi-analytical sensitivity computation," *Engineering Analysis with Boundary Elements*, vol. 31, no. 4, pp. 343–360, 2007.

[112] Santosa, F., "A level-set approach for inverse problems involving obstacles," *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 1, pp. 17–33, 1996.

[113] Seitz, S. and Baker, S., "Filter flow," in *12th International Conference on Computer Vision*, pp. 143–150, IEEE, 2009.

[114] Sethian, J., *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, vol. 3. Cambridge University Press, 1999.

[115] Shen, T., Zhang, S., Huang, J., Huang, X., and Metaxas, D., "Integrating shape and texture in 3D deformable models: from metamorphs to active volume models," in *Multi Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies*, pp. 1–31, Springer, 2011.

[116] Singer, I. and Turkel, E., "High-order finite difference methods for the Helmholtz equation," *Computer methods in applied mechanics and engineering*, vol. 163, no. 1, pp. 343–358, 1998.

[117] Staib, L. and Duncan, J., "Boundary finding with parametrically deformable models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, 1992.

[118] Tai, X.-C. and Chan, T., "A survey on multiple level set methods with applications for identifying piecewise constant functions," *International Journal of Numerical Analysis and Modeling*, vol. 1, no. 1, pp. 25–47, 2004.

[119] Thomas, J., *Numerical Partial Differential Equations: Finite Difference Methods*, vol. 22. Springer, 1995.

[120] Toreyin, B., Dedeoglu, Y., Gudukbay, U., and Cetin, A., "Computer vision based method for real-time fire and flame detection," *Pattern Recognition Letters*, vol. 27, no. 1, pp. 49–58, 2006.

[121] Tsai, A., Yezzi, A., Wells, W., Tempany, C., Tucker, D., Fan, A., Grimson, E., and Willsky, A., "A shape-based approach to the segmentation of medical imagery using level sets," *IEEE Transactions on Medical Imaging*, vol. 22, no. 2, pp. 137–154, 2003.

[122] van den Berg, P. and Kleinman, R., "A contrast source inversion method," *Inverse Problems*, vol. 13, no. 6, p. 1607, 1997.

[123] Van Den Doel, K., Ascher, U., and Leitao, A., "Multiple level sets for piecewise constant surface reconstruction in highly ill-posed problems," *Journal of Scientific Computing*, vol. 43, no. 1, pp. 44–66, 2010.

[124] Vedaldi, A. and Fulkerson, B., "VLFeat: An open and portable library of computer vision algorithms." `http://www.vlfeat.org/`, 2008.

[125] Verstockt, S., Vanoosthuyse, A., Van Hoecke, S., Lambert, P., and Van de Walle, R., "Multi-sensor fire detection by fusing visual and non-visual flame features," *Image and Signal Processing*, pp. 333–341, 2010.

[126] Werlberger, M., Pock, T., and Bischof, H., "Motion estimation with non-local total variation regularization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2464–2471, IEEE, 2010.

[127] Werlberger, M., Trobin, W., Pock, T., Wedel, A., Cremers, D., and Bischof, H., "Anisotropic Huber-L1 optical flow," in *Proceedings of the British machine vision conference*, vol. 34, pp. 1–11, 2009.

[128] Yang, J. and Duncan, J., "3D image segmentation of deformable objects with joint shape-intensity prior models using level sets," *Medical Image Analysis*, vol. 8, no. 3, pp. 285–294, 2004.

[129] Yezzi, A., Tsai, A., and Willsky, A., "A statistical approach to snakes for bimodal and trimodal imagery," in *International Conference on Computer Vision (ICCV)*, vol. 2, pp. 898–903, IEEE, 1999.

[130] Zacharopoulos, A., Arridge, S., Dorn, O., Kolehmainen, V., and Sikora, J., "Three-dimensional reconstruction of shape and piecewise constant region values for optical tomography using spherical harmonic parametrization and a boundary element method," *Inverse Problems*, vol. 22, no. 5, p. 1509, 2006.

[131] Zhao, J., Zhang, Z., Han, S., Qu, C., Yuan, Z., and Zhang, D., "SVM based forest fire detection using static and dynamic features," *Computer Science and Information Systems*, no. 00, pp. 30–30, 2011.

[132] Zienkiewicz, O. and Taylor, R., *The Finite Element Method*, vol. 3. McGraw-Hill, London, 1977.

[133] Zienkiewicz, O., Taylor, R., and Zhu, J., *The Finite Element Method: Its Basis and Fundamentals*, vol. 1. Butterworth-Heinemann, 2005.